

Федеральное агентство по образованию

---

Государственное образовательное учреждение  
высшего профессионального образования  
Санкт Петербургский государственный технологический институт  
(Технический университет)

---

Кафедра систем автоматизированного проектирования и управления

**Н.П. Жадановская**



## **Операционные системы**

Методические указания к выполнению контрольных работ  
для студентов заочной формы обучения  
направления подготовки «Информатика и вычислительная техника»

Санкт–Петербург  
2010

## УДК 681.3.066

Жадановская, Н.П. Операционные системы: методические указания к выполнению контрольных работ / Н.П. Жадановская – СПб.: СПбГТИ(ТУ), 2010. –120 с.

В методические указания включены задания для выполнения контрольных работ, предназначенных для изучения фундаментальных концепций и принципов построения, справедливых для большинства известных на сегодня операционных систем.

Каждая контрольная работа обязательно включает практическое задание, позволяющее оценить полученные знания при исследовании функционирования современных операционных систем на примере операционных систем семейства MSWindows. Это может быть как MSWindows XP, так и другая версия системы.

Методические указания предназначены для студентов 3 курса заочной формы обучения направления подготовки 230100 «Информатика и вычислительная техника» и соответствуют разделам рабочей программы дисциплины «Операционные системы».

Ил. 59, табл. 22, библиогр. назв. 10

Рецензент:

1 Г.А. Соколов, д-р техн. наук, профессор,  
зав.кафедрой автоматизации процессов химической  
промышленности СПбГТИ(ТУ)

Утверждены на заседании учебно-методической комиссии факультета информатики и управления

Рекомендовано к изданию РИСо СПбГТИ(ТУ)

© СПбГТИ(ТУ), 2010г.

## СОДЕРЖАНИЕ

|  |            |
|--|------------|
| <b>ВВЕДЕНИЕ</b> .....                              | <b>4</b>   |
| <b>КОНТРОЛЬНАЯ РАБОТА №1.</b>                      |            |
| ОРГАНИЗАЦИЯ ФАЙЛОВОЙ СИСТЕМЫ .....                 | <b>6</b>   |
| <b>КОНТРОЛЬНАЯ РАБОТА №2.</b>                      |            |
| ОРГАНИЗАЦИЯ И УПРАВЛЕНИЕ ОСНОВНОЙ<br>ПАМЯТЬЮ ..... | <b>41</b>  |
| <b>КОНТРОЛЬНАЯ РАБОТА №3.</b>                      |            |
| УПРАВЛЕНИЕ ПРОЦЕССАМИ .....                        | <b>85</b>  |
| <b>ЛИТЕРАТУРА</b> .....                            | <b>119</b> |

## ВВЕДЕНИЕ

Сегодня существует большое количество разных типов операционных систем, отличающихся областями применения, аппаратными платформами и методами реализации. Естественно, это обуславливает и значительные функциональные различия этих операционной системы.

Даже у конкретной операционной системы набор выполняемых функций зачастую определить не так просто - та функция, которая сегодня выполняется внешним по отношению к системе компонентом, завтра может стать ее неотъемлемой частью и наоборот.

Поэтому при изучении операционных систем очень важно из всего многообразия выделить те функции, которые присущи всем операционным системам как классу продуктов.

В данном учебном пособии представлены задания на выполнение контрольных работ по изучению фундаментальных концепций и принципов построения, справедливых для большинства известных сегодня операционных систем.

Дисциплина «Операционные системы» принадлежит к числу общепрофессиональных (ОПД).

Цель дисциплины – дать представление об основных понятиях и терминах операционных систем, а также дать теоретические основы построения, организации и функционирования современных операционных систем для персональных компьютеров и научить управлять функционированием персональным компьютером и отдельными его устройствами, оценивать параметры выбираемых операционных систем, конфигурировать, оптимизировать и настраивать вычислительную систему для решения конкретных классов задач.

Дисциплина «Операционные системы» для студентов заочного отделения читается на 3 курсе. Студенты решают три контрольные работы и сдают экзамен.

В учебном пособии составлены три контрольные работы, включающие в себя содержание работы, варианты и примеры решения.

Каждая контрольная работа содержит три части: аналитическую, тестовую и практическую.

Проведенный аналитический обзор должен помочь выполнить остальные части контрольной работы.

Тестовые задания должны показать качество проработки теоретических вопросов.

Практические задания позволяют применить полученные знания не только для решения теоретических задач, но и для использования встроенных специальных средств современной операционной системы, рассчитанных на пользователя профессионала.

В Примерах решения к каждой контрольной работе можно найти определенные указания для выполнения практического задания по использованию штатных средств рекомендуемой операционной системы.

Студент самостоятельно выбирает задания контрольной работы. Номер варианта соответствует номеру первой буквы фамилии студента согласно таблице 1.

**Таблица 1** - Распределение вариантов заданий

| <b>Первая буква фамилии студента</b>     | <b>Номер варианта</b> | <b>Первая буква фамилии студента</b> | <b>Номер варианта</b> |
|--|-----------------------|--------------------------------------|-----------------------|
| <b>А</b>                                 | <b>1</b>              | <b>Н, О</b>                          | <b>9</b>              |
| <b>Б, В</b>                              | <b>2</b>              | <b>П, Р</b>                          | <b>10</b>             |
| <b>Г, Д</b>                              | <b>3</b>              | <b>С, Т</b>                          | <b>11</b>             |
| <b>Е, Ё</b>                              | <b>4</b>              | <b>У, Ф</b>                          | <b>12</b>             |
| <b>Ж, З</b>                              | <b>5</b>              | <b>Х, Ц</b>                          | <b>13</b>             |
| <b>И, Й</b>                              | <b>6</b>              | <b>Ч, Ш</b>                          | <b>14</b>             |
| <b>К</b>                                 | <b>7</b>              | <b>Щ, Э</b>                          | <b>15</b>             |
| <b>Л, М</b>                              | <b>8</b>              | <b>Ю, Я</b>                          | <b>16</b>             |
| <b>Пример решения контрольной работы</b> |                       |                                      | <b>17</b>             |

Студенту необходимо представить отчёт о выполненных контрольных работах в распечатанном виде и в электронном виде на любом носителе информации. Отчёт должен включать: титульный лист, условие задачи и предложенное решение поставленных задач. На титульном листе отчёта о выполнении контрольных работ необходимо указать фамилию, имя и отчество студента, номер учебной группы, номер контрольной работы, номер варианта.

Приступая к выполнению контрольных работ, рекомендуется также ознакомиться с методическими материалами и литературой, предложенной в списке рекомендуемой литературы.

# 1 КОНТРОЛЬНАЯ РАБОТА №1

## Тема 1 Организация файловой системы

Основные функции файловой системы. Способы организации файлов. Объединение в блоки и буферизация. Виды файлов и способы организации информации в файлах (последовательные, прямого доступа, индексные, библиотечные и т.д.). Способы организации файловой структуры. Справочники и списки как основа файловой структуры. Распределение внешней памяти: Связное распределение и Несвязное распределение (цепочки блоков, цепочки индексных блоков, таблицы отображения блоков). Целостность файловой системы. Иерархия подсистем файловой системы. Методы доступа и управление доступом. Защита от несанкционированного доступа.

### Содержание работы:

#### Задание №1 – Аналитический обзор

Напишите ответ на поставленный вопрос. Ответ на вопросы желательно давать в развернутом виде. Вы должны показать глубину ваших знаний в ваших ответах. Эти знания помогут также выполнить остальные части контрольной работы.

#### Задание №2 – Тестовые задания

2.1 Вставьте пропущенное слово

2.2 Выберите правильные ответы (ответ м.б. не один) на предложенное тестовое задание

#### Задание №3 – Практические задания

3.1 Предложите решение теоретической задачи.

3.2 С использованием встроенных средств современной операционной системы получите полную информацию о состоянии жестких дисков своего компьютера и проанализируйте ее.

### Указания

Выполнение практического задания 3.2 очень важно.

При обслуживании информационных систем большое значение имеет качество используемых диагностических средств. В состав современных операционных систем обязательно включаются средства для устранения возникающих проблем по работе с жесткими дисками.

У пользователей м.б. свои любимые утилиты различных фирм производителей относящиеся к информационно – диагностическому программному обеспечению.

В операционную систему Windows XP для возможности оптимизации системы для работы с логическими дисками включены:

- Проверка диска - проверяет жесткие диски на предмет ошибок файловой системы – потерянных и пересекающихся кластеров и т.д.
- Дефрагментация диска - приводит в порядок данные на жестком диске для повышения производительности, т.к. при большой фрагментации файлов замедляется работа системы.
- Очистка диска - автоматически удаляет ненужные файлы, освобождая место на диске, тем самым сокращая долю адресного пространства, которое необходимо обрабатывать системе.
- Мастер обслуживания - позволяет автоматизировать запуск задач оптимизации системы.

Теоретические знания, полученные в данном разделе курса, позволят понять многие параметры этих системных программ и оптимизировать работу вашей вычислительной системы.

Для получения доступа к просмотру и состоянию и управлению дисками необходимо выбрать значок **Мой компьютер**, затем выбрать строку **Управление** и в открывшемся окне выбрать строку **Управление дисками**.

В данной операционной системе имеется мощный центр справки и поддержки, который также позволяет найти нужные процедуры (и пояснения по ним), если их нет в списке программ. Здесь также можно получить сведения об использовании жесткого диска, а также другого оборудования.

В **Примере решения** (Вариант 17) также дано более подробное руководство для выполнения этого практического задания.

Приступая к выполнению контрольных работ, рекомендуется ознакомиться со следующими методическими материалами:

1. Жадановская Н.П. Операционные системы. Базовый курс [Текст]: учебное пособие для студентов заочной формы обучения. – СПб.: СПбГТИ(ТУ), 2009. –xxx с.
2. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. Учебник для ВУЗов. - СПб.: ПИТЕР, 2007.- 544с.
3. Microsoft Windows XP Home Edition и Professional/ Русские версии/ Под общей редакцией А.Н. Чекмарева. – СПб.: БХВ-Петербург, 2006. - 640с.

В качестве руководства пользователя по операционной системе и Professional м.б. использовано любое популярное руководство пользователя.

## Контрольная работа 1: Вариант 1

### 1 Аналитическая часть

Перечислите способы организации файловых структур.

### 2 Тестовые задания

**2.1** Архитектура подсистемы ввода-вывода позволяет достаточно просто включать в ее состав новые типы ..... систем, без необходимости переписывания кода.

**2.2** Разметку диска под конкретный тип файловой системы выполняют процедуры высокоуровневого или логического форматирования. Полученный при этом размер логического диска:

- меньше или равен размеру раздела
- равен размеру раздела
- больше или равен размеру раздела
- равен сумме всех разделов

### 3 Практические задания

**3.1** На рисунке В1.1 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 2              | 0   |     | 8  |     | 16 | 6   | 24 |     |
| B         | ...      | 15             | 1   |     | 9  |     | 17 |     | 25 | 26  |
| C         | ...      | 30             | 2   | 3   | 10 | 16  | 18 | 19  | 26 | 27  |
| D         | ...      | 18             | 3   | 4   | 11 | bad | 19 | 13  | 27 | 28  |
|           |          |                | 4   | 5   | 12 | eof | 20 |     | 28 | eof |
|           |          |                | 5   | eof | 13 | 12  | 21 | bad | 29 |     |
|           |          |                | 6   | eof | 14 | 13  | 22 |     | 30 | 10  |
|           |          |                | 7   |     | 15 | 14  | 23 |     | 31 |     |

Рисунок В1.1 – Фрагмент файловой системы

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.



## Контрольная работа 1: Вариант 2

### 1 Аналитическая часть

Уточните способы организации информации в файлах и перечислите виды файлов, имеющих разную организацию.

### 2 Тестовые задания

2.1 Данные на дисках организуются в ..... системы, свойства которых во многом определяют свойства самой операционной системы - ее отказоустойчивость, быстродействие, максимальный объем хранимых данных.

2.2 Файловая система является частью:

- дисковых систем
- драйверов дисков
- операционной системы
- пользовательских программ
- систем управления основной памятью

### 3 Практические задания

3.1 На рисунке В1.2 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 26             | 0   |     | 8  | bad | 16 | 17  | 24 | 12  |
| B         | ...      | 16             | 1   |     | 9  | 19  | 17 | 12  | 25 |     |
| C         | ...      | 2              | 2   | 15  | 10 | 11  | 18 | bad | 26 | 27  |
| D         | ...      | 23             | 3   |     | 11 | 12  | 19 | eof | 27 | 28  |
|           |          |                | 4   | 5   | 12 | 13  | 20 |     | 28 | 29  |
|           |          |                | 5   | 6   | 13 | eof | 21 |     | 29 | eof |
|           |          |                | 6   | 7   | 14 |     | 22 |     | 30 |     |
|           |          |                | 7   | eof | 15 | 9   | 23 | 24  | 31 |     |

Рисунок В1.2 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 3

### 1 Аналитическая часть

Поясните, что понимается под связным распределением внешней памяти.

### 2 Тестовые задания

2.1 Прежде чем форматировать диск под определенную файловую систему, он должен быть разбит на .....

2.2 Минимальная единица информации, участвующая в операциях обмена с дисковым устройством:

- символ
- бит
- байт
- сектор
- дорожка
- цилиндр

### 3 Практические задания

3.1 На рисунке В1.3 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 11             | 0   |     | 8  |     | 16 |     | 24 |     |
| B         | ...      | 2              | 1   |     | 9  |     | 17 | bad | 25 | 26  |
| C         | ...      | 22             | 2   | 3   | 10 | 11  | 18 |     | 26 | eof |
| D         | ...      | 19             | 3   | 25  | 11 | 12  | 19 | 14  | 27 |     |
|           |          |                | 4   | 6   | 12 | 13  | 20 |     | 28 | eof |
|           |          |                | 5   | bad | 13 | eof | 21 | 25  | 29 | 28  |
|           |          |                | 6   | eof | 14 | 4   | 22 | 21  | 30 | 29  |
|           |          |                | 7   |     | 15 |     | 23 |     | 31 | 30  |

Рисунок В1.3 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 4

### 1 Аналитическая часть

Поясните, что зависит от размера блока при блочном распределении внешней памяти.

### 2 Тестовые задания

**2.1** При удалении файла из файловой системы FAT в первый байт соответствующей записи ..... заносится признак, свидетельствующий о том, что эта запись свободна, а во все индексные указатели файла заносится признак «кластер свободен».

**2.2** Для подготовки жесткого диска к работе сначала выполняется процедура физического или низкоуровневого форматирования. Низкоуровневое форматирование создает:

- разделы
- секторы
- корневой каталог
- таблицу отображения файлов
- кластеры

### 3 Практические задания

**3.1** На жестком диске использует размещение файлов в виде связанного списка кластеров. Ссылка на кластер занимает на жестких дисках 4 байта. Запись каталога вместе с атрибутами файла занимает 32 байта. Расход емкости диска на загрузочную запись не учитывается и считается, что последний кластер файла заполнен в среднем на 60%.

Необходимо определить:

- максимальное количество файлов, которое можно разместить на диске,
- долю адресной информации в процентах от емкости диска
- долю неиспользованной емкости диска за счет внутренней фрагментации.

Считая, что:

- емкость диска - 40 Гбайт,
- размер кластера 4 Кбайт.

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 5

### 1 Аналитическая часть

Поясните, что происходит при выполнении процедуры физического, или низкоуровневого, форматирования диска, предшествующей его использованию.

### 2 Тестовые задания

2.1 Еще один способ задания физического расположения файла заключается в простом перечислении номеров ....., занимаемых этим файлом

2.2 Существует несколько разновидностей файловых систем FAT. Что означает числовое значение 16 и 32 в системах FAT16 и FAT32:

- разрядность операционной системы
- размер кластера на диске
- разрядность элемента в таблице FAT
- допустимое количество символов в имени файла
- размер записи элемента каталога

### 3 Практические задания

3.1 На рисунке В1.5 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 15             | 0   |     | 8  | 9   | 16 |     | 24 | 15  |
| B         | ...      | 23             | 1   |     | 9  | 10  | 17 | 18  | 25 |     |
| C         | ...      | 12             | 2   |     | 10 | eof | 18 | eof | 26 | bad |
| D         | ...      | 29             | 3   | 4   | 11 |     | 19 |     | 27 |     |
|           |          |                | 4   | eof | 12 | 13  | 20 |     | 28 | 29  |
|           |          |                | 5   |     | 13 | 17  | 21 |     | 29 | 30  |
|           |          |                | 6   | bad | 14 |     | 22 | 17  | 30 | 31  |
|           |          |                | 7   | 8   | 15 | 3   | 23 | 22  | 31 | eof |

Рисунок В1.5 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 6

### 1 Аналитическая часть

Перечислите основные стратегии, используемые при планировании работ с дисковой памятью.

### 2 Тестовые задания

2.1 Пользователь должен иметь удобные средства работы с файлами, включая ....., объединяющие файлы в группы, средства поиска файлов по признакам и набор команд для создания, модификации и удаления файлов.

2.2 Что означает «1024» в элементе таблицы FAT для кластера:

- кластер свободен;
- кластер используется файлом и не является последним,
- последний кластер файла;
- дефектный кластер;
- резервный кластер.

### 3 Практические задания

3.1 На рисунке В1.6 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |    |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|----|----|-----|----|-----|
| A         | ...      | 22             | 0   |     | 8  |    | 16 | 17  | 24 | 25  |
| B         | ...      | 20             | 1   |     | 9  |    | 17 | eof | 25 | eof |
| C         | ...      | 10             | 2   |     | 10 | 11 | 18 |     | 26 |     |
| D         | ...      | 5              | 3   | bad | 11 | 6  | 19 |     | 27 | bad |
|           |          |                | 4   |     | 12 |    | 20 | 21  | 28 | eof |
|           |          |                | 5   | 13  | 13 | 29 | 21 | 6   | 29 | 28  |
|           |          |                | 6   | 7   | 14 | 15 | 22 | 23  | 30 |     |
|           |          |                | 7   | eof | 15 | 16 | 23 | 24  | 31 |     |

Рисунок В1.6 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 7

### 1 Аналитическая часть

Поясните, что происходит при выполнении процедур высокоуровневого, или логического, форматирования диска, предшествующих его использованию.

### 2 Тестовые задания

**2.1** Очевидно, что надежно можно восстановить только файлы, которые были расположены в последовательных ..... диска.

**2.2** При подготовке жесткого диска к работе необходимо выполнить определенные процедуры. Укажите порядок операций при работе с новым жестким диском, на котором устанавливаются системы семейства MS Windows:

- высокоуровневое форматирование
- низкоуровневое форматирование
- создание логических дисков
- создание разделов

### 3 Практические задания

**3.1** На рисунке В1.7 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 11             | 0   |     | 8  |     | 16 | 27  | 24 | 3   |
| B         | ...      | 2              | 1   |     | 9  |     | 17 |     | 25 |     |
| C         | ...      | 22             | 2   |     | 10 | 11  | 18 | eof | 26 | eof |
| D         | ...      | 19             | 3   | 31  | 11 | 12  | 19 | 20  | 27 | 26  |
|           |          |                | 4   |     | 12 | 13  | 20 | 21  | 28 |     |
|           |          |                | 5   | bad | 13 | eof | 21 | 22  | 29 |     |
|           |          |                | 6   | 7   | 14 |     | 22 | eof | 30 | bad |
|           |          |                | 7   | 27  | 15 | 16  | 23 |     | 31 | 18  |

Рисунок В1.7 – Фрагмент файловой системы

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 8

### 1 Аналитическая часть

Поясните, что понимается под несвязным распределением внешней памяти.

### 2 Тестовые задания

**2.1** Файловая система представляет собой комплекс системных программных средств, реализующих различные операции с ....., таких как создание, уничтожение, чтение, запись, именование и поиск.

**2.2** Для подготовки жесткого диска к работе сначала выполняется процедура физического или низкоуровневого форматирования. Какой разметки нет на диске:

- дорожек
- кластеров
- цилиндров
- секторов
- разделов
- каталогов

### 3 Практические задания

**3.1** На жестком диске использует размещение файлов в виде связанного списка кластеров. Ссылка на кластер занимает на жестких дисках 4 байта. Запись каталога вместе с атрибутами файла занимает 32 байта. Расход емкости диска на загрузочную запись не учитывается и считается, что последний кластер файла заполнен в среднем на 50%.

Необходимо определить:

- максимальное количество файлов, которое можно разместить на диске,
- долю адресной информации в процентах от емкости диска
- долю неиспользованной емкости диска за счет внутренней фрагментации.

Считая, что:

- емкость диска - 80 Гбайт,
- размер кластера 16 Кбайт.

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 9

### 1 Аналитическая часть

С учетом многослойного подхода к построению программного обеспечения покажите иерархию подсистем файловой системы.

### 2 Тестовые задания

2.1 ..... устанавливают соответствие между именами файлов и их характеристиками, используемыми файловой системой для управления файлами, такими как тип файла и расположение его на диске,

2.2 Минимальный фактический размер файла на жестком диске равен:

- 1 биту
- 1 байту
- 1 символу
- 1 сектору
- 1 кластеру

### 3 Практические задания

3.1 На рисунке В1.9 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 5              | 0   |     | 8  |     | 16 | 17  | 24 | 25  |
| B         | ...      | 31             | 1   |     | 9  | bad | 17 | 18  | 25 | eof |
| C         | ...      | 16             | 2   | eof | 10 | eof | 18 | 19  | 26 |     |
| D         | ...      | 14             | 3   |     | 11 | 10  | 19 | eof | 27 |     |
|           |          |                | 4   | 11  | 12 |     | 20 |     | 28 | 7   |
|           |          |                | 5   | 4   | 13 |     | 21 | bad | 29 |     |
|           |          |                | 6   |     | 14 | 28  | 22 | 23  | 30 | 11  |
|           |          |                | 7   |     | 15 |     | 23 | 24  | 31 | 30  |

Рисунок В1.9 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.



## Контрольная работа 1: Вариант 10

### 1 Аналитическая часть

Поясните, в чем состоят преимущества и недостатки блочного распределения внешней памяти.

### 2 Тестовые задания

2.1 Логическая запись является наименьшим элементом данных, которым может оперировать программист при организации обмена с внешним устройством.

2.2 На диске не может быть кластера размером:

- 512 байт
- 1024 байта
- 1536 байт
- 2048 байт
- 32768 байт

### 3 Практические задания

3.1 На рисунке В1.10 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof – последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      |                | 0   |     | 8  | 9   | 16 | 17  | 24 |     |
| B         | ...      |                | 1   |     | 9  | 10  | 17 | 3   | 25 | bad |
| C         | ...      |                | 2   |     | 10 | 11  | 18 | 13  | 26 |     |
| D         | ...      |                | 3   | 4   | 11 | eof | 19 | 20  | 27 | 6   |
|           |          |                | 4   | eof | 12 |     | 20 | 21  | 28 |     |
|           |          |                | 5   |     | 13 | 27  | 21 | 22  | 29 | 30  |
|           |          |                | 6   | eof | 14 | bad | 22 | eof | 30 | 3   |
|           |          |                | 7   |     | 15 |     | 23 |     | 31 |     |

Рисунок В1.10 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 11

### 1 Аналитическая часть

Поясните, что понимается под целостностью файловой структуры.

### 2 Тестовые задания

2.1 Файловая система распределяет дисковую память, поддерживает именование ....., отображает их имена в соответствующие адреса во внешней памяти и обеспечивает доступ к данным, а также поддерживает их разделение, защиту и восстановление.

2.2 Что означает «512» в элементе таблицы FAT для некоторого кластера:

- кластер свободен;
- кластер используется файлом и не является последним,
- последний кластер файла;
- дефектный кластер;
- резервный кластер.

### 3 Практические задания

3.1 На рисунке В1.11 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 4              | 0   |     | 8  |     | 16 | bad | 24 | 31  |
| B         | ...      | 9              | 1   |     | 9  | 6   | 17 | 18  | 25 |     |
| C         | ...      | 20             | 2   |     | 10 |     | 18 | 28  | 26 |     |
| D         | ...      | 28             | 3   |     | 11 | 12  | 19 |     | 27 |     |
|           |          |                | 4   | 5   | 12 | 13  | 20 | 21  | 28 | 29  |
|           |          |                | 5   | 28  | 13 | 14  | 21 | 22  | 29 | eof |
|           |          |                | 6   | 24  | 14 | eof | 22 | 23  | 30 |     |
|           |          |                | 7   | bad | 15 |     | 23 | eof | 31 | eof |

Рисунок В1.11 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 12

### 1 Аналитическая часть

Поясните, в чем состоят преимущества и недостатки непрерывного (связного) распределения внешней памяти.

### 2 Тестовые задания

**2.1** Операционная система может поддерживать разные ..... разделов, особым образом отмечая разделы, которые могут быть использованы для загрузки модулей операционной системы, и разделы, в которых можно устанавливать только приложения и хранить файлы данных.

**2.2** Разметку диска под конкретный тип файловой системы выполняют процедуры высокоуровневого или логического форматирования. Операционные системы семейства Windows поддерживают следующие типы разделов:

- основной
- подкачки
- дополнительный
- расширенный
- сжатый

### 3 Практические задания

**3.1** На жестком диске использует размещение файлов в виде связного списка кластеров. Ссылка на кластер занимает на жестких дисках 4 байта. Запись каталога вместе с атрибутами файла занимает 32 байта. Расход емкости диска на загрузочную запись не учитывается и считается, что последний кластер файла заполнен в среднем на 55%.

Необходимо определить:

- максимальное количество файлов, которое можно разместить на диске,
- долю адресной информации в процентах от емкости диска
- долю неиспользованной емкости диска за счет внутренней фрагментации.

Считая, что:

- емкость диска - 120 Гбайт,
- размер кластера 32 Кбайт.

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера.

С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы Windows XP, предоставляемые пользователю, для работы с жесткими дисками.

## Контрольная работа 1: Вариант 13

### 1 Аналитическая часть

Поясните, в чем разница понятий физического сектора, логического сектора и кластера

### 2 Тестовые задания

2.1 Достоинством подсистемы ввода-вывода любой универсальной операционной системы является наличие разнообразного набора ..... для наиболее популярных периферийных устройств.

2.2 Раздел, с которого загружается операционная система при запуске компьютера, называется:

- загрузочным
- основным
- активным
- системным

### 3 Практические задания

3.1 На рисунке В1.13 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 6              | 0   |     | 8  | bad | 16 |     | 24 |     |
| B         | ...      | 11             | 1   |     | 9  | 14  | 17 |     | 25 | 26  |
| C         | ...      | 25             | 2   |     | 10 | 3   | 18 | 19  | 26 | 27  |
| D         | ...      | 30             | 3   | 4   | 11 | 10  | 19 | 20  | 27 | 28  |
|           |          |                | 4   | eof | 12 |     | 20 | 21  | 28 | eof |
|           |          |                | 5   |     | 13 |     | 21 | eof | 29 | 3   |
|           |          |                | 6   | 23  | 14 | eof | 22 | bad | 30 | 29  |
|           |          |                | 7   |     | 15 |     | 23 | 9   | 31 |     |

Рисунок В1.13 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии своего жесткого диска и поясните полученные результаты. С учетом размера кластера и логической конфигурации жесткого диска своего компьютера поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации. Оцените возможности штатных средств операционной системы по оптимизации работы с жесткими дисками.

## Контрольная работа 1: Вариант 14

### 1 Аналитическая часть

Расскажите о справочниках и списках, являющихся основными структурами данных для организации файловых систем.

### 2 Тестовые задания

2.1 В подсистеме ввода-вывода для согласования скоростей обмена также широко используется ..... данных в оперативной памяти.

2.2 На диске не может быть кластера размером:

- 512 байт
- 1024 байта
- 8200 байт
- 16320 байт
- 32768 байт

### 3 Практические задания

3.1 На рисунке В14 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 3              | 0   |     | 8  | 9   | 16 | 28  | 24 |     |
| B         | ...      | 6              | 1   |     | 9  | 10  | 17 | bad | 25 | 26  |
| C         | ...      | 14             | 2   | 3   | 10 | 11  | 18 |     | 26 | 19  |
| D         | ...      | 25             | 3   | 4   | 11 | eof | 19 | 20  | 27 |     |
|           |          |                | 4   | 5   | 12 |     | 20 | eof | 28 | 22  |
|           |          |                | 5   | eof | 13 |     | 21 |     | 29 | bad |
|           |          |                | 6   | 7   | 14 | 16  | 22 | eof | 30 |     |
|           |          |                | 7   | 19  | 15 |     | 23 |     | 31 |     |

Рисунок В1.14 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера и поясните полученные результаты. С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы по оптимизации работы с жесткими дисками.

## Контрольная работа 1: Вариант 15

### 1 Аналитическая часть

Поясните, что понимается под физической и логической записью файла.

### 2 Тестовые задания

**2.1** Дорожки и секторы создаются в результате выполнения процедуры ..... форматирования диска, предшествующей использованию диска.

**2.2** Разметку диска под конкретный тип файловой системы выполняют процедуры высокоуровневого или логического форматирования. Операционные системы семейства Windows поддерживают следующие типы разделов:

- основной
- подкачки
- дополнительный
- расширенный

### 3 Практические задания

**3.1** На рисунке В1.15 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |   |   |    |     |    |     |    |    |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|---|---|----|-----|----|-----|----|----|
| A         | ...      | 3              | 0   |     | 8  | 9   | 16 | 17  | 24 |     | 1 |   | 9  | eof | 17 | 18  | 25 | 26 |
| B         | ...      | 15             | 2   | eof | 10 |     | 18 | 19  | 26 | eof | 3 | 4 | 11 | 2   | 19 | eof | 27 |    |
| C         | ...      | 16             | 4   | 25  | 12 | bad | 20 | bad | 28 | 21  | 5 |   | 13 |     | 21 | 11  | 29 |    |
| D         | ...      | 28             | 6   | 7   | 14 | 15  | 22 |     | 30 |     | 7 | 8 | 15 | 25  | 23 |     | 31 |    |

**Рисунок В1.15** – Фрагмент файловой системы

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии своего жесткого диска и поясните полученные результаты. С учетом размера кластера и логической конфигурации жесткого диска своего компьютера поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности этой операционной системы по оптимизации работы с жесткими дисками.

## Контрольная работа 1: Вариант 16

### 1 Аналитическая часть

Уточните проблемы, решаемые при организации доступа к файлам.

### 2 Тестовые задания

2.1 Операционные системы разного типа используют единое для всех них представление о ....., но создают на его основе логические устройства, специфические для каждого типа ОС.

2.2 Одна файловая система занимает, как правило:

- 1 физический диск
- 1 логический диск
- 1 раздел диска
- 1 таблицу отображения файлов
- 1 файл подкачки

### 3 Практические задания

3.1 На рисунке В1.16 приведен пример фрагмента файловой системы MS Windows FAT\* – содержимое каталога и таблицы FAT. Проанализируйте целостность данной файловой системы и дайте рекомендации по ее лечению и оптимизации считая, что eof - последний кластер файла, bad – дефектный кластер и не заполненный элемент таблицы – свободный кластер.

| Имя файла | Атрибуты | Номер кластера | FAT |     |    |     |    |     |    |     |
|-----------|----------|----------------|-----|-----|----|-----|----|-----|----|-----|
| A         | ...      | 2              | 0   |     | 8  |     | 16 |     | 24 | 25  |
| B         | ...      | 5              | 1   |     | 9  | 22  | 17 | eof | 25 | 19  |
| C         | ...      | 24             | 2   | 3   | 10 | 11  | 18 |     | 26 | 27  |
| D         | ...      | 26             | 3   | 19  | 11 | 12  | 19 | 20  | 27 | 28  |
|           |          |                | 4   |     | 12 | 13  | 20 | eof | 28 | 29  |
|           |          |                | 5   | 9   | 13 | eof | 21 |     | 29 | eof |
|           |          |                | 6   |     | 14 |     | 22 | 17  | 30 |     |
|           |          |                | 7   | bad | 15 |     | 23 |     | 31 | bad |

Рисунок В1.16 – Фрагмент файловой системы

3.2 Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии жесткого диска своего компьютера и поясните полученные результаты. С учетом размера кластера и логической конфигурации жесткого диска поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы по оптимизации работы с жесткими дисками.

## Контрольная работа 1: Вариант 17 - Примеры решения

### 1 Аналитическая часть

Перечислите основные функции файловой системы.

### 2 Тестовые задания

**2.1** Для уменьшения потерь дискового пространства из-за внутренней фрагментации желательно ..... делать небольшими, а для сокращения объема адресной информации и как следствие - повышения скорости обмена с дисками - наоборот.

**2.2** В файловых системах FAT в операционных системах семейства MS Windows атрибуты файлов хранятся:

- в таблице разделов
- вместе с файлом
- в каталогах
- в индексных дескрипторах
- в файле подкачки
- в таблицах FAT

### 3 Практические задания

**3.1** На жестком диске используется размещение файлов в виде связанного списка кластеров. Ссылка на кластер занимает на жестких дисках 4 байта. Запись каталога вместе с атрибутами файла занимает 32 байта. Расход емкости диска на загрузочную запись не учитывается и считается, что последний кластер файла заполнен в среднем на 70%.

Необходимо определить:

- максимальное количество файлов, которое можно разместить на диске,
- долю адресной информации в процентах от емкости диска
- долю неиспользованной емкости диска за счет внутренней фрагментации.

Считая, что:

- емкость диска - 20 Гбайт,
- размер кластера 8 Кбайт.

**3.2** Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии своего жесткого диска и поясните полученные результаты. С учетом размера кластера и логической конфигурации жесткого диска своего компьютера поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.

Оцените возможности операционной системы по оптимизации работы с жесткими дисками.



## Контрольная работа 1: Вариант 17 - Примеры решения

### 1 Аналитическая часть

**Напишите ответ на поставленный вопрос**

**Перечислите основные функции файловой системы.**

Для большинства пользователей файловая система - наиболее видимая часть **операционной системы - ОС**. Она предоставляет механизм для хранения и доступа, как к данным, так и к программам для всех пользователей системы.

**Файловая система** - это часть **ОС**, назначение которой состоит в том, чтобы организовать эффективную работу с данными, хранящимися во внешней памяти, и обеспечить пользователю удобный интерфейс при работе с такими данными.

Непосредственное взаимодействие с диском - прерогатива компонента системы ввода-вывода **ОС**, называемого драйвером диска.

Файловая система для пользователя обеспечивает **следующие возможности:**

- обращение к файлам по символьным именам (т.е. без имен физических устройств) - т.е. обеспечение независимости файлов от устройств;
- создания, модификации и уничтожения файлов - т.е. предоставление средств манипуляции файлами;
- задания и использования удобных структур файлов для прикладных программ и организация передачи информации между файлами;
- сохранения и восстановления информации;
- разделения файлов друг от друга при общей работе (при тщательном контроле **ОС**), при этом механизм для разделения должен предусматривать различные варианты контроля доступа (для чтения, для записи, для выполнения);
- защиты информации от несанкционированного доступа (файловая система может предусматривать даже шифровку данных);
- не задумываться об именах физических устройств, о форматах данных или физических средствах обмена данными с этими устройствами, т.е. файловая система должна иметь "дружественный интерфейс" для пользователя.

Т.о. файловая система решает как вопросы структуры, именования, защиты файлов; операции, которые разрешается производить над файлами, так и проблемы выделения дискового пространства, обеспечения производительной работы файловой системы и т.д.

Иногда говорят, что файл - это поименованный набор связанной информации, записанной во вторичную память. Правила именования файлов зависят от **ОС**. Многие **ОС** поддерживают имена из двух частей (имя + расширение). Тип расширения файла позволяет **ОС** организовать работу с ним различных программ в соответствии с заранее оговоренными соглашениями. Обычно **ОС** накладывают ограничения, как на используемые в имени символы, так и на длину имени файла.

Основные типы файлов: регулярные (обычные) файлы и директории (справочники, каталоги). Обычные файлы содержат пользовательскую информацию. Директории - системные файлы, поддерживающие структуру файловой системы. В каталоге содержится перечень входящих в него файлов и устанавливается соответствие между файлами и их характеристиками (атрибутами).

Все файлы имеют некоторую внутреннюю структуру. Способ организации информации в файле может оказать существенное влияние на стоимость хранения, доступа и использования файла. Под организацией файлов понимают способ расположения записей файла во внешней памяти или логическую структуру записей файла и взаимосвязи между ними. ОС поддерживают несколько вариантов структуризации файлов:

- Последовательный файл
- Файл прямого доступа

ОС не осуществляет никакой интерпретации содержимого файла. Эта схема обеспечивает максимальную гибкость и универсальность.

Кроме имени ОС связывают с каждым файлом характеристики файлов - атрибутами. Список атрибутов в разных ОС может варьироваться, но обычно он содержит: основную информацию (имя, тип файла), адресную информацию (устройство, начальный адрес, размер), информацию об управлении доступом (владелец, допустимые операции) и информацию об использовании (даты создания, последнего чтения, модификации и др.). Список атрибутов обычно хранится в структуре директорий.

ОС предоставляет набор операций для работы с файлами. При работе с файлом обычно выполняется не одна, а несколько операций: найти данные файла и его атрибуты по символьному имени, считать необходимые атрибуты файла в отведенную область оперативной памяти и проанализировать права пользователя на выполнение требуемой операции. Затем выполнить операцию, после чего освободить занимаемую данными файла область памяти.

Основные операции над файлами:

- Создание файла
- Удаление файла
- Открытие файла. Открытие файла перед его использованием является процедурой создания дескриптора или управляющего блока файла.
- Закрытие файла. Если работа с файлом завершена, то файл нужно закрыть, чтобы освободить место в таблицах файловой системы.
- Чтение данных из файла.
- Запись данных в файл

Есть и другие операции, например переименование файла, получение атрибутов файла и т. д.

ОС предоставляет также набор операций для работы с директориями, и хотя директории - это файлы, логика работы с ними имеет отличия:

- Создание директории.
- Удаление директории.
- Открытие директории.

- Заккрытие директории.
- Поиск в директории.
- Получение списка файлов в каталоге.
- Переименование.
- Создание файла. При создании нового файла необходимо добавить в каталог соответствующий элемент.
- Удаление файла. Удаление из каталога соответствующего элемента.

Задание пути к файлу в файловых системах некоторых ОС отличается тем, с чего начинается эта цепочка имен. В современных ОС принято разбивать диски на логические диски, называемые разделами. В некоторых системах требуется, чтобы каждый архив файлов целиком располагался на одном логическом диске и тогда полное имя файла начинается с буквы накопителя.

Итак, файловая система - это совокупность файлов и взаимосвязей между ними. Главная же функция файловой системы - распределение пространства внешней памяти, а следовательно и выделение и освобождение места в дисковой памяти, т.е она отвечает не только за способы организации файлов, а и за способы организации файловых структур.

Алгоритмы выделения дискового пространства и способы учета свободной и занятой дисковой памяти в разных файловых системах различны. Однако для каждого из методов запись в директории, соответствующая символному имени файла, всегда содержит указатель, следуя которому можно найти все блоки данного файла.

В файловых системах используются две основные стратегии:

- Связное распределение внешней памяти.
- Несвязное (блочное) распределение внешней памяти.

Существует несколько распространенных способов реализации систем поблочного распределения:

- цепочки блоков,
- цепочки индексных блоков;
- таблицы отображения блоков.

Каждый алгоритм выделения внешней памяти порождает необходимость в процедурах, учитывающих особенности алгоритма и обеспечивающих проверку и сохранение целостности файловой системы, а также повышение производительности.

Кроме этого, для обеспечения надежности информация д.б. защищена от несанкционированного доступа, поэтому в ОС при наличии в системе многих пользователей организуется контроль доступа к файлам - операции: чтения, записи и выполнения. Общий подход к защите файлов основан на организации доступа, зависящего от идентификатора пользователя. Для этого в системах создают классификации пользователей, например: Владелец, Группа (пользователи, нуждающихся в типовом способе доступа к файлу) и Остальные.

Итак, файловая система представляет собой набор файлов, директорий и операций над ними. Имена, структуры файлов, способы доступа к ним и их

атрибуты - важные аспекты организации файловой системы. Главная задача файловой системы - связать символическое имя файла с данными на диске. Большинство современных ОС поддерживает иерархическую систему директорий с возможным вложением. Безопасность файловой системы, базируется на ведении списков прав доступа.

## 2 Тестовые задания

### 2.1 Вставьте пропущенное слово

Для уменьшения потерь из-за внутренней фрагментации желательно ..... делать небольшими, а для сокращения объема адресной информации и повышения скорости обмена с дисками наоборот.

Правильным вариантом ответа на поставленный вопрос является: **кластеры (блоки)**. При блочной организации файловых структур файлу выделяется целое количество блоков - объединение секторов. Блок в ОС семейства MSWindows называется кластером. Необходимо отметить, что даже при отсутствии внешней фрагментации на уровне кластеров на диске все равно имеется определенное количество областей памяти, которые невозможно использовать, то есть существует внутренняя фрагментация. Эти фрагменты представляют собой неиспользуемые части последних кластеров, назначенных файлам, поскольку объем файла в общем случае не кратен размеру кластера. На каждом файле в среднем теряется половина кластера. Это потери особенно велики, когда на диске имеется большое количество маленьких файлов, а кластер имеет большой размер. Размеры кластеров зависят от размера раздела и типа файловой системы. Примерный диапазон, в котором может меняться размер кластера, составляет от 512 байт – 1 сектор до десятков килобайт.

### 2.2 Выберите все правильные ответы на поставленный вопрос:

**В файловых системах FAT в операционных системах семейства Windows атрибуты файлов хранятся:**

- в таблице разделов
- вместе с файлом
- в каталогах
- в индексных дескрипторах
- в файле подкачки
- в таблицах FAT

Правильным вариантом ответа на поставленный вопрос является: **в каталогах**. Набор атрибутов файла определяется спецификой файловой системы: в файловых системах разного типа для характеристики файлов могут использоваться разные наборы атрибутов.

Пользователь может получать доступ к атрибутам, используя средства, предоставленные для этих целей файловой системой. Обычно разрешается читать значения любых атрибутов, а изменять - только некоторые.

Значения атрибутов файлов могут непосредственно содержаться в каталогах, как это сделано в файловой системе FAT в **операционной системе – ОС** - семейства Windows. В этом случае запись в каталоге содержит простое символьное имя и атрибуты файла: R — только для чтения, A — архивный, H — скрытый, S — системный (здесь буквами обозначены признаки файла).

### 3 Практические задания

#### 3.1 Предложите решение теоретической задачи.

На жестком диске использует размещение файлов в виде связного списка кластеров. Ссылка на кластер занимает на жестких дисках 4 байта. Запись каталога вместе с атрибутами файла занимает 32 байта. Расход емкости диска на загрузочную запись не учитывается и считается, что последний кластер файла заполнен в среднем на 70%.

Необходимо определить:

- максимальное количество файлов, которое можно разместить на диске,
- долю адресной информации в процентах от емкости диска
- долю неиспользованной емкости диска за счет внутренней фрагментации.

Считая, что:

- емкость диска - 20 Гбайт,
- размер кластера 8 Кбайт.

Для данной задачи предлагается следующее решение:

К адресной информации файла относятся запись в каталоге с учетом атрибутов файла  $A_a$  - 32 байта и запись для ссылки на очередной кластер в цепочке  $A_c$  – 4 байта.

Так как требуется определить максимальное количество файлов на диске, то файлу необходимо выделить хотя бы один кластер, т.к. меньше кластера файлу выделить невозможно.

Таким образом адресная информация на один файл  $A_f$  составит :

$$A_f = A_a + A_c = 32 + 4 = 36 \text{ байт}$$

Количество искомых файлов –  $K_f$ .

Общее количество адресной информации  $A$  будет следующим:

$$A = K_f * A_f$$

Остальная часть диска может быть занята файлами, каждый из которых имеет минимальный размер, т. е. размер одного кластера –  $A_k = 8$  Кбайта.

Итак емкость диска  $E_d$  равна адресной информации на все файлы плюс все файлы размером один кластер:

$$E_d = K_f * A_f + K_f * A_k = K_f (A_f + A_k)$$

а максимальное количество файлов равно:

$$K = E_d / (A_f + A_k)$$

Итак, **максимальное количество файлов, которое можно разместить на диске равно** (размер строки каталога измеряется в байтах, а диска в

Гбайтах, поэтому все единицы необходимо привести к байтам, а количество м.б. только целым):

$$K = 20 * 2^{30} / (36 + 2^{13}) = 20 * 1073741824 / (8192 + 36) = 21474836480 / 8228 = \mathbf{2609970}$$

объем адресной информации составляет:

$$A = A_f * 2609970 = 36 * 2609970 = 93958920 \text{ байт}$$

**доля адресной информации в процентах от емкости диска равна:**

$$A\% = 100\% * A / E_d = 93958920 / 20 * 1073741824 = \mathbf{0,437\%}$$

Для определения доли неиспользованной емкости диска за счет внутренней фрагментации считаем, что последний кластер файла заполнен в среднем на 70%. Т.к. каждому файлу выделяется только один кластер, то последним кластером является каждый выделенный кластер, поэтому **доля потерь дискового пространства Еп составит:**

$$E_p\% = 100\% * E_p / E_d = 100\% * 0.3 * 8192 * 2609970 / 20 * 1073741824 = \mathbf{29,86\%}$$

### **3.2 Выполните предложенное практическое задание**

**Используя штатные средства операционной системы MS Windows XP получите отчет о состоянии своего жесткого диска и поясните полученные результаты.**

**С учетом размера кластера на логическом диске С: своего компьютера поясните полученные значения свободного и занятого пространства и оцените возможные потери дискового пространства из-за внутренней фрагментации.**

**Оцените возможности операционной системы по оптимизации работы с жесткими дисками.**

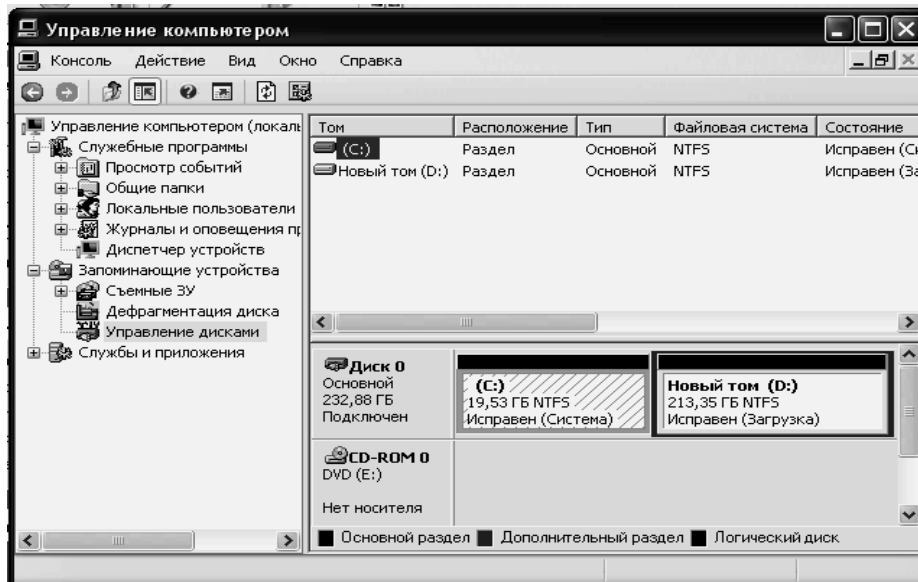
В качестве тестовой системы для выполнения практического задания использовалась: **операционная система – ОС - Windows XP PRO SP3.**

При обслуживании информационных систем большое значение имеет качество используемых диагностических средств. В состав современных ОС обязательно включаются средства для устранения **возникающих** проблем по работе с дисками.

В ОС Windows XP для возможности оптимизация системы по работе с логическими дисками есть:

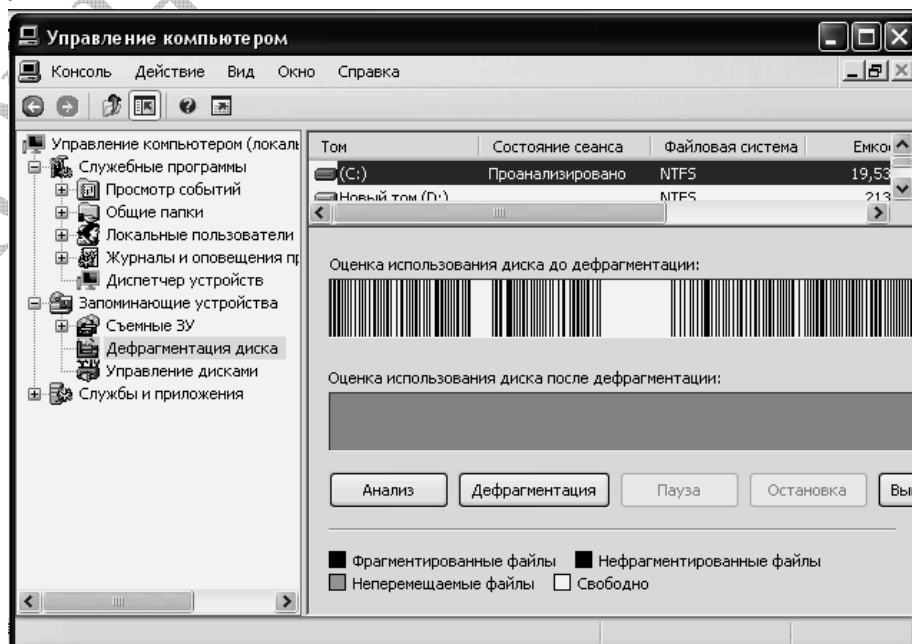
- Проверка диска - проверяет жесткие диски на предмет ошибок файловой системы – потерянных и пересекающихся кластеров и т.д.
- Дефрагментация диска - приводит в порядок данные на жестком диске для повышения производительности, т.к при большой фрагментации файлов замедляется работа системы..
- Очистка диска - автоматически удаляет ненужные файлы, освобождая место на диске, тем самым сокращая долю адресного пространства, которое необходимо обрабатывать системе.
- Мастер обслуживания - позволяет автоматизировать запуск задач оптимизации системы.

С использованием штатных средств ОС есть возможность просматривать состояние и управлять дисками. На рисунке В.1.17.1 показано состояние и все параметры дисковых устройств моего компьютера: количество разделов на жестком диске, типы разделов, количество логических дисков и установленные файловые системы.



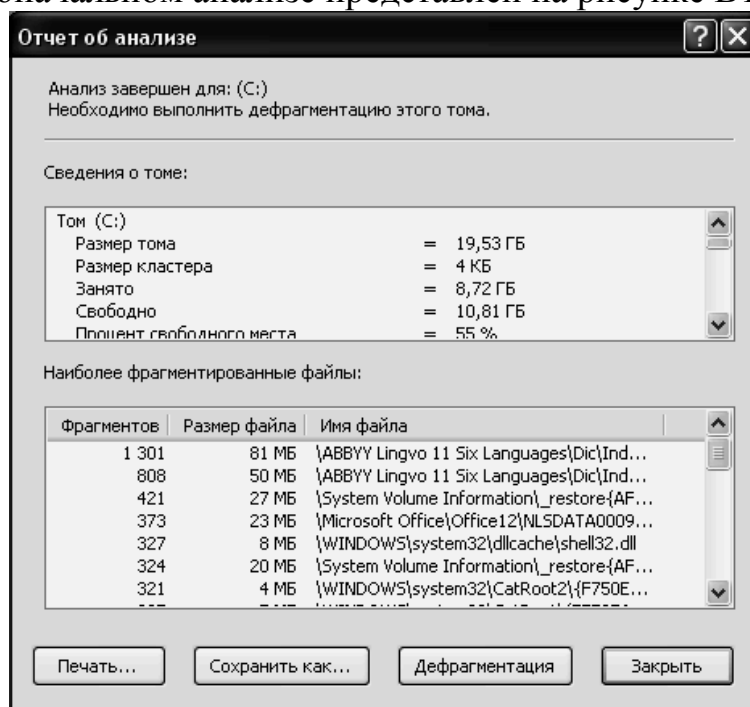
**Рисунок В1.17.1 -** Возможность управления дисками

При работе с жестким диском всегда имеет место фрагментация. Часто файл бывает разрезанным на сотни фрагментов, разбросанных по всему жесткому диску. И хотя фрагментация не является ошибкой файловой системы, а особенностью блочной организации файловых систем, необходимо периодически ее уменьшать. Возможность для этого в системе показана на рисунке В1.17.2



**Рисунок В1.17.2 –** Первоначальный анализ диска

Отчет о первоначальном анализе представлен на рисунке В1.17.3



**Рисунок В1.17.3** – Первоначальный анализ диска

Подробный анализ с перечнем наиболее фрагментированных файлов можно получить в файл, содержимое которого приведено ниже.

Итак, в файле отчета содержится:

Том (C:)

|                          |            |
|--------------------------|------------|
| Размер тома              | = 19,53 ГБ |
| Размер кластера          | = 4 КБ     |
| Занято                   | = 8,72 ГБ  |
| Свободно                 | = 10,81 ГБ |
| Процент свободного места | = 55 %     |

Фрагментация тома

|                               |        |
|-------------------------------|--------|
| Всего фрагментировано         | = 31 % |
| Фрагментация файлов           | = 60 % |
| Фрагментация свободного места | = 2 %  |

Фрагментация файлов

|                              |          |
|------------------------------|----------|
| Всего файлов                 | = 87 042 |
| Средний размер файла         | = 158 КБ |
| Всего фрагментировано файлов | = 14 949 |
| Всего лишних фрагментов      | = 66 576 |
| В среднем фрагментов на файл | = 1,76   |

Фрагментация файла подкачки



Размер файла подкачки = 0 байт  
 Всего фрагментов = 0

Фрагментация папок

Всего папок = 6 627  
 Фрагментировано папок = 88  
 Лишних фрагментов папок = 580

Фрагментация MFT (Master File Table)

Общий размер MFT = 140 МБ  
 Счетчик записей MFT = 94 015  
 Процент использования MFT = 65 %  
 Всего фрагментов MFT = 3

---

| Фрагментов | Размер файла | Наиболее фрагментированные файлы  |
|------------|--------------|---|
| 1301       | 81 МБ        | \ABBY Lingvo 11 Six Languages\Dic\Index\1_deposit.dat   |
| 808        | 50 МБ        | \ABBY Lingvo 11 Six Languages\Dic\Index\1_coord.htf   |
| 421        | 27 МБ        | \System Volume Information\_restore{AF4EBBBF-057B-4536-A1E0-6D4CFF3D8008}\RP141\A0078489.dll                        |
| 373        | 23 МБ        | \Microsoft Office\Office12\NLSDATA0009.DLL  |
| 327        | 8 МБ         | \WINDOWS\system32\dllcache\shell32.dll  |
| 324        | 20 МБ        | \System Volume Information\_restore{AF4EBBBF-057B-4536-A1E0-6D4CFF3D8008}\RP136\A0077832.exe                        |
| 321        | 4 МБ         | \WINDOWS\system32\CatRoot2\{F750E6C3-38EE-11D1-85E5-00C04FC295EE}\catdb   |
| 307        | 5 МБ         | \WINDOWS\system32\CatRoot\{F750E6C3-38EE-11D1-85E5-00C04FC295EE}\sp3_uber.cat                                       |
| 289        | 46 МБ        | \WINDOWS\Downloaded Installations\{E2A3BA4B-E704-42B6-AB10-8251332323E2}\Sony Ericsson PC Suite 1.20.173.msi        |
| 289        | 46 МБ        | \WINDOWS\Downloaded Installations\{C5ADA65A-7828-4D85-B071-ECC52B51F794}\Sony Ericsson PC Suite 1.20.173.msi        |
| 261        | 16 МБ        | \ABBY Lingvo 11 Six Languages\Dic\Index\1_deposit.ind   |
| 254        | 16 МБ        | \Program Files\Common Files\Microsoft Shared\OFFICE12\MSO.DLL   |
| 252        | 16 МБ        | \WINDOWS\repair\software  |
| 234        | 15 МБ        | \Microsoft Office\OFFICE11\1049\PUB60.CTG   |
| 229        | 15 МБ        | \Program Files\Common Files\System\Ole DB\msmdlocal.dll   |
| 228        | 15 МБ        | \Program Files\Realtek\InstallShield\RTHDCPL.exe  |
| 191        | 13 МБ        | \Microsoft Office\Office12\OART.DLL   |
| 189        | 27 МБ        | \System Volume Information\_restore{AF4EBBBF-057B-4536-A1E0-6D4CFF3D8008}\RP126\snapshot\_REGISTRY_MACHINE_SOFTWARE |
| 173        | 8 МБ         | \WINDOWS\\$\hf_mig\$\KB967715\SP2QFE\shell32.dll  |
| 170        | 11 МБ        | \Microsoft Office\Office12\VISLIB.DLL   |
| 159        | 8 МБ         | \WINDOWS\\$\NtUninstallKB967715\$\shell32.dll.000   |
| 159        |              | 636 КБ  |
|            |              | \WINDOWS\SoftwareDistribution\Download\eced8b5ea8e636fb8bff2b719fa62647   |
| 157        | 10 МБ        | \Program Files\Common Files\Microsoft Shared\PROOF\MSGR3GE.LEX  |
| 156        | 18 МБ        | \WINDOWS\Installer\37221d.msp   |
| 148        | 11 МБ        | \Program Files\OLYMPUS\OLYMPUS Master\OLYMPUSMaster.pdf   |

Как видно из отчета логический диск нуждается в дефрагментации. Эти рекомендации дает и сама ОС. Данная процедура довольно длительная на больших дисках. На рисунке В1.17.4 виден процесс удаления фрагментации на диске.

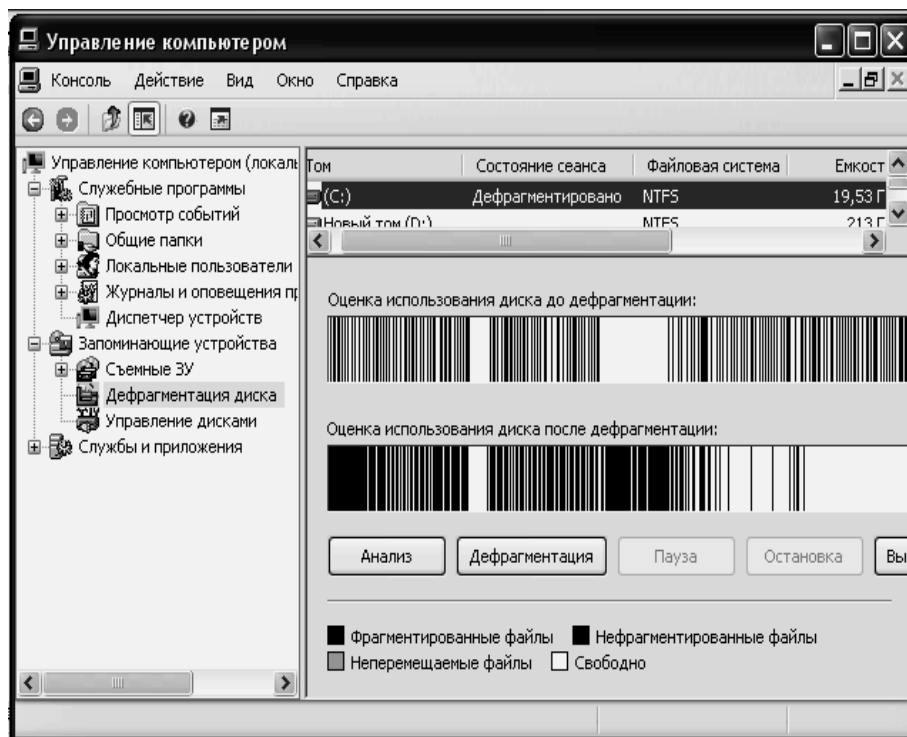


Рисунок В1.17.4 – Выполнение дефрагментации логического диска

Представленный отчет на рисунке В1.17.5 показывает результат проведенной дефрагментации.

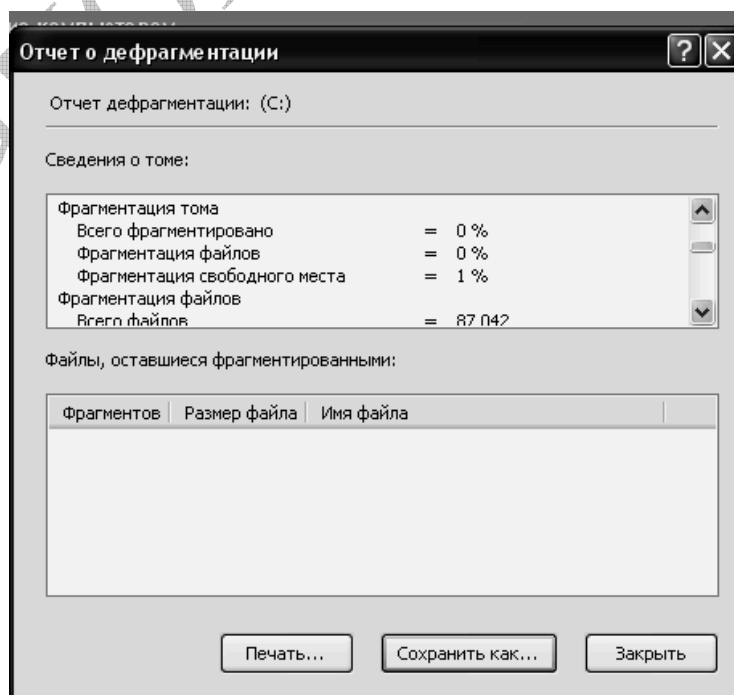


Рисунок В1.17.5 - Результат проведенной дефрагментации.

Результат дефрагментации полностью виден и из отчета, содержимое которого приведено ниже. Итак, в файле отчета содержится:

Том (С:)

|                          |            |
|--------------------------|------------|
| Размер тома              | = 19,53 ГБ |
| Размер кластера          | = 4 КБ     |
| Занято                   | = 8,72 ГБ  |
| Свободно                 | = 10,81 ГБ |
| Процент свободного места | = 55 %     |

Фрагментация тома

|                               |       |
|-------------------------------|-------|
| Всего фрагментировано         | = 0 % |
| Фрагментация файлов           | = 0 % |
| Фрагментация свободного места | = 1 % |

Фрагментация файлов

|                              |          |
|------------------------------|----------|
| Всего файлов                 | = 87042  |
| Средний размер файла         | = 158 КБ |
| Всего фрагментировано файлов | = 0      |
| Всего лишних фрагментов      | = 0      |
| В среднем фрагментов на файл | = 0,99   |

Фрагментация файла подкачки

|                       |          |
|-----------------------|----------|
| Размер файла подкачки | = 0 байт |
| Всего фрагментов      | = 0      |

Фрагментация папок

|                         |        |
|-------------------------|--------|
| Всего папок             | = 6627 |
| Фраgmentировано папок   | = 1    |
| Лишних фрагментов папок | = 0    |

Фрагментация MFT (Master File Table)

|                           |          |
|---------------------------|----------|
| Общий размер MFT          | = 140 МБ |
| Счетчик записей MFT       | = 94015  |
| Процент использования MFT | = 65 %   |
| Всего фрагментов MFT      | = 3      |

---

Фрагментов    Размер файла    Файлы, которые не могут быть дефрагментированы  
отсутствует

Внешняя фрагментация убрана, однако при блочном распределении внешней памяти есть и внутренняя фрагментация. Последний кластер каждого файла заполнен в среднем на 50%.

Из приведенного отчета очень хорошо видны потери дискового пространства из-за внутренней фрагментации.

Всего файлов = 87042  
 Размер кластера = 4 КБ  
 Размер потерь =  $87042 * 4 / 2$  = 174084 КБ

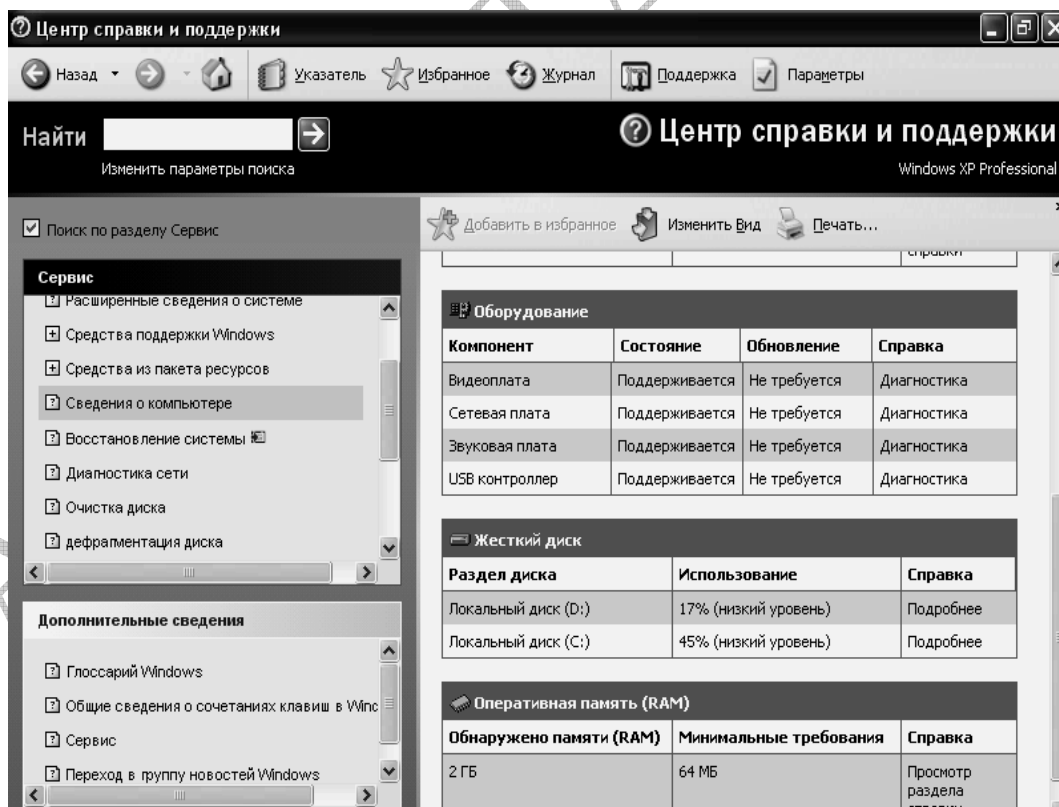
Учитывая

- небольшой размер логического диска = 19,53 Гб
- процент свободного места = 55 %

потери довольно большие.

Размер кластера – величина, влияющая на потери из-за внутренней фрагментации. Есть возможность отформатировать логический диск с меньшим размером кластера, например 512б. Однако при этом увеличиваются объемы адресной информации, которые влияют на производительность. Т.е размер кластера – компромиссная задача.

В данной ОС имеется мощный центр справки и поддержки, который тоже позволяет найти нужные процедуры, если их нет в списке программ. Здесь также можно получить сведения об использовании жесткого диска, а также другого оборудования. На рисунок В1.17.6 показаны дополнительные возможности контроля используемого оборудования в том числе и дисков.



**Рисунок В1.17.6** – Дополнительные характеристики оборудования

Здесь также можно вызвать и процедуру дефрагментацию, да еще и получить справку по этой процедуре (см. рисунок В1.17.7)

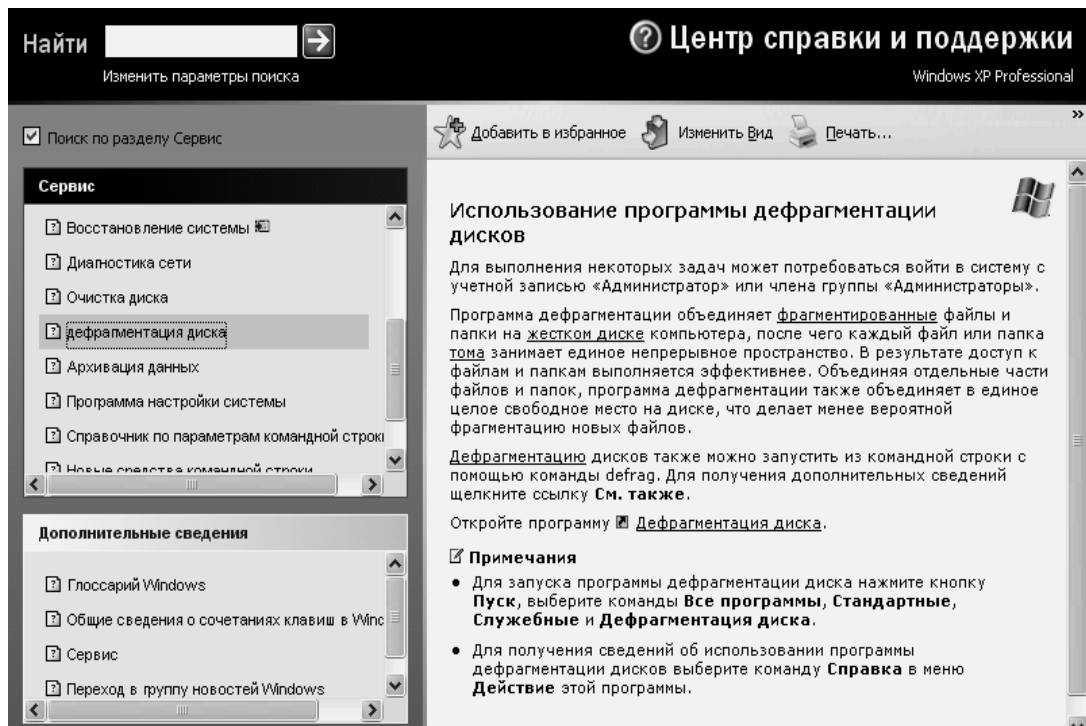


Рисунок В1.17.7 – Вызов процедуры очистки диска

На рисунке В1.17.8 показана возможность вызова процедуры очистки диска.

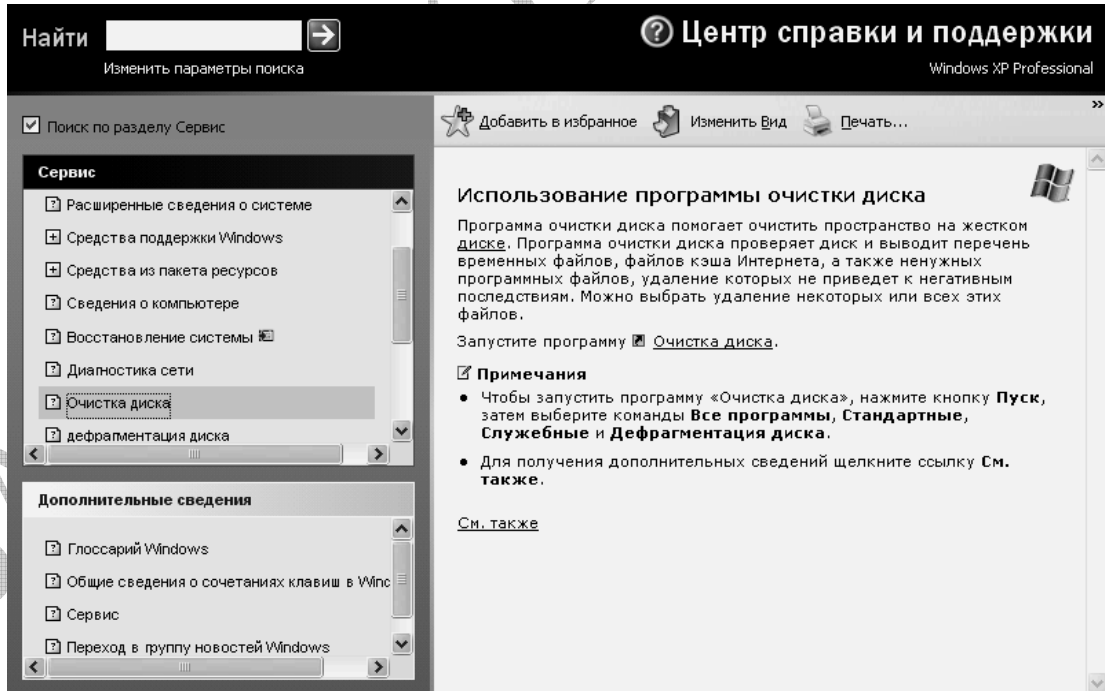


Рисунок В1.17.8 – Вызов процедуры очистки диска

При очистке диска есть возможность получить прогноз результата будущей операции (см. рисунок В1.17.9), выбрать файлы для возможного

удаления (см. рисунок В1.17.10) и даже увеличить свободное место за счет элементов системы (см. рисунок В1.17.11).

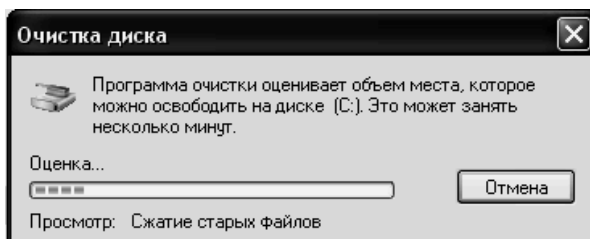


Рисунок В1.17.9 – Прогноз очистки диска



Рисунок В1.17.10 – Выбор типов файлов для удаления

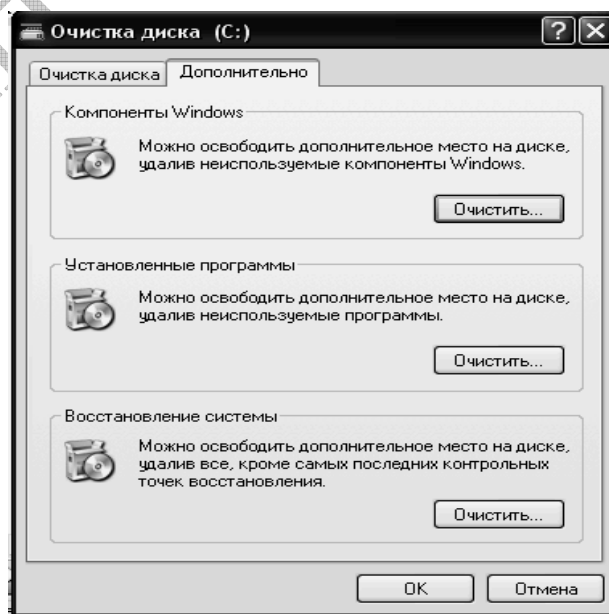


Рисунок В1.17.11 – Профессиональная очистка диска

Целостность файловой системы – также важная составляющая нормальной работы вычислительной системы. Ошибки файловой системы появляются в перезагрузки системы в результате зависания системы, а когда это не дает результатов и отключения питания. Все файловые системы в данной ОС – блочные. Типичные ошибки этой организации – потерянные и пересекающиеся кластеры. При аварийной перезагрузке программы контроля целостности могут запуститься перед загрузкой ОС.

На рисунке В1.17.12 показана возможность вызова команды ОС, выполняющую эту процедуру. Ход и результат выполнения этой операции представлен на рисунке В1.17.13

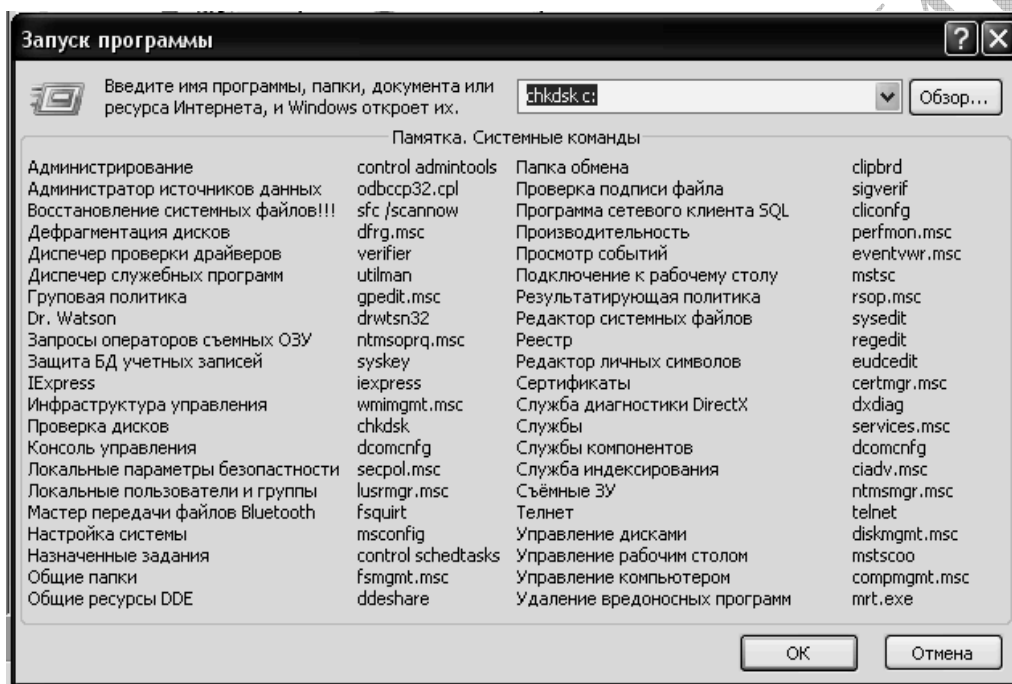


Рисунок В1.17.12 – Использование команды ОС

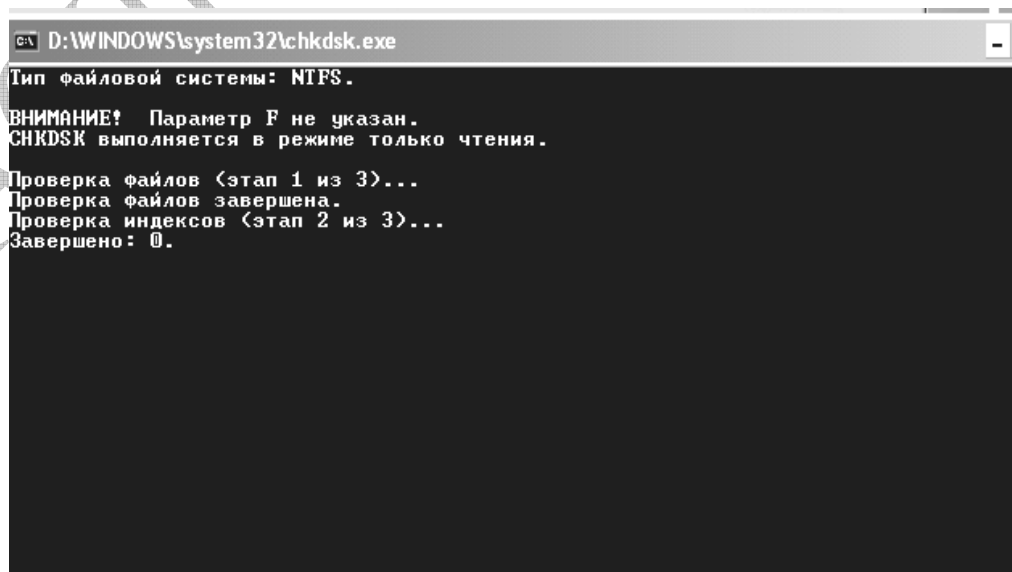


Рисунок В1.17.13 - Результат выполнения команды ОС chkdsk

Несмотря на представленные средства ОС 100% гарантии сохранности информации не существует. В связи с этим все ОС обязательно имеют в своем составе средство создания резервных копий системных данных. На рисунке В1.17.14 показано наличие этих возможностей и на рисунке В1.17.15 способ выполнения этих действий.

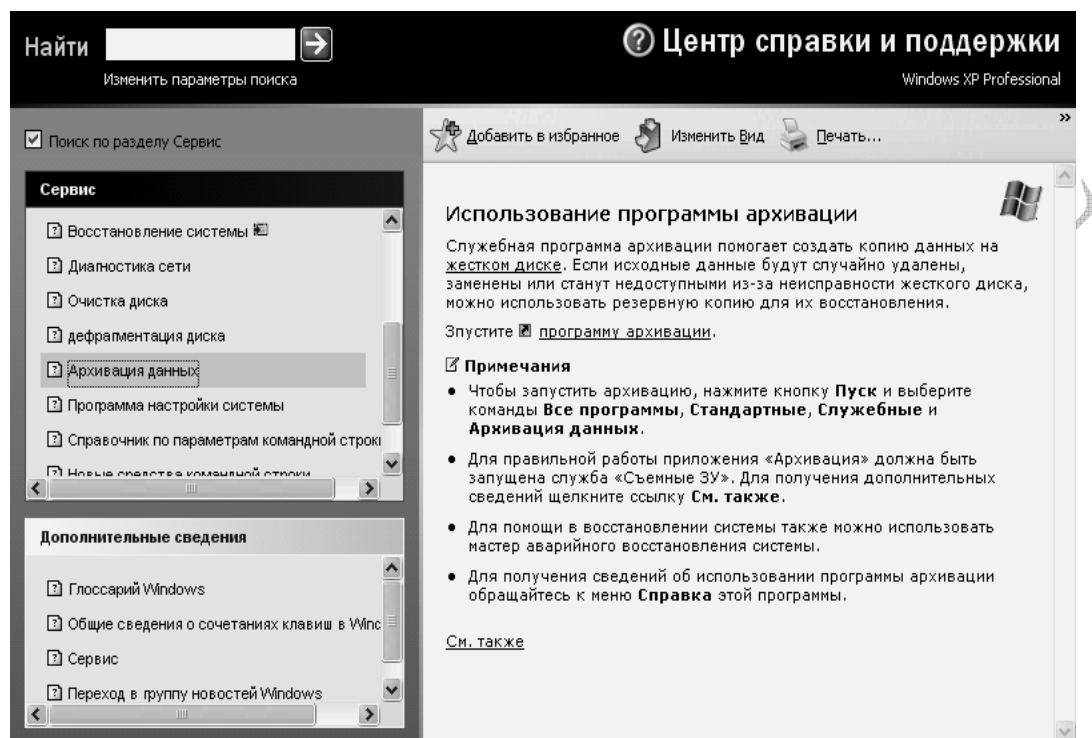


Рисунок В1.17.14 – Копирование данных – обязательная операция

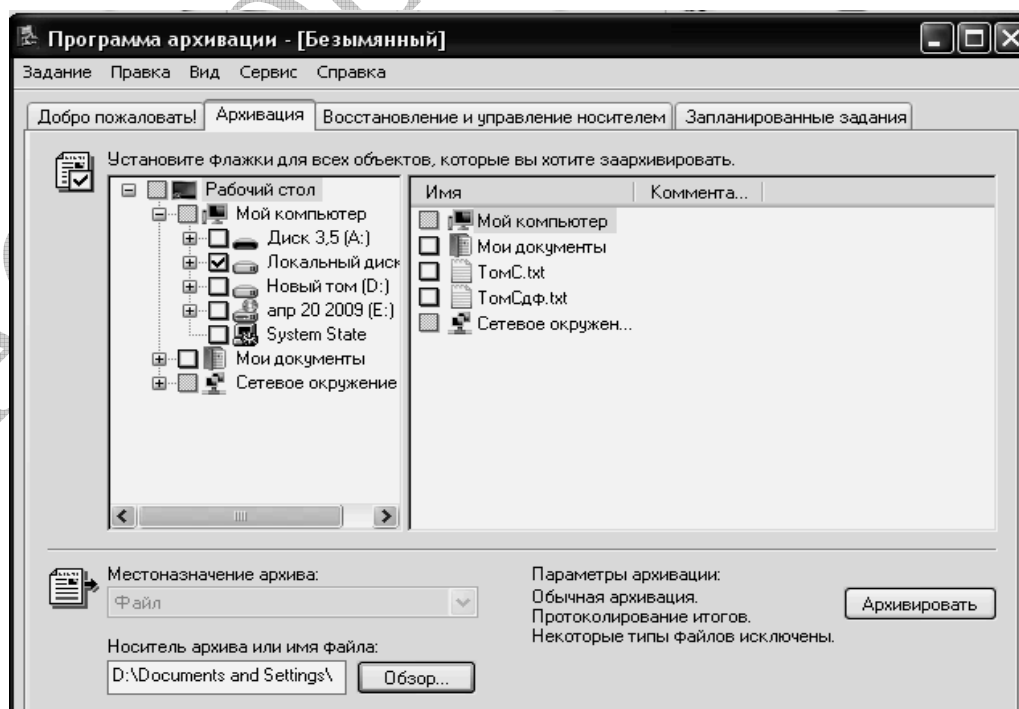


Рисунок В1.17.15 – Возможности программы архивации



## 2 КОНТРОЛЬНАЯ РАБОТА №2

### Тема: Управление памятью.

Распределение основной памяти: задача распределения основной памяти; особенности основной памяти как ресурса ВС. Алгоритмы распределения основной памяти. Оверлейная структура. Мультипрограммирование с фиксированными и переменными разделами. Защита памяти. Виртуальная память. Страничная организация памяти: таблицы страниц. Сегментная организация виртуальной памяти; преобразование адресов; организация таблиц сегментов. Сегментно-страничная организация: преобразование адресов, особенности реализации. Коллективное использование информации и защита информации для страничной, сегментной и сегментно-страничной организации памяти. Ассоциативная память как средство поддержки виртуальной памяти. Стратегии управления виртуальной памятью. Схемы динамического преобразования адресов. Алгоритмы "выталкивания" страниц: выталкивание случайной страницы, FIFO, LRU, LFU. Анализ размера страницы. Рабочее множество: определение, стратегия рабочего множества. Анализ страничного распределения. Исключительные ситуации при работе с памятью. Thrashing. Аномалия FIFO (Биледи).

### Содержание работы:

#### Задание №1 – Аналитический обзор

Напишите ответ на поставленный вопрос. Ответ на вопросы желательно давать в развернутом виде. Вы должны показать глубину ваших знаний в ваших ответах. Эти знания помогут также выполнить остальные части контрольной работы.

#### Задание №2 – Тестовые задания

2.1 Вставьте пропущенные слова

2.2 Выберите правильные ответы (ответ м.б. не один) на предложенные тестовые задания.

#### Задание №3 – Практические задания

3.1 Предложите решение теоретической задачи.

3.2 С использованием встроенных средств современной операционной системы получите полную информацию о компонентах вычислительного процесса, которые касаются деятельности подсистемы управления памятью, и проанализируйте ее.

Исследование этих компонент вычислительного процесса можно проводить на примере самостоятельно разработанных вами программ по курсу «Программирование на языках высокого уровня» или на приложениях, активно используемых вами на рабочем месте.

## Указания

В операционной системе MS Windows XP такими штатными средствами являются диспетчер задач и системный монитор. Эти средства необходимы для анализа работоспособности системы.

Теоретические знания, полученные в данном разделе курса, позволят понять многие параметры этих системных программ и оптимизировать работу вашей вычислительной системы, выбирая оптимальные параметры для управления памятью, файлом подкачки и т.д.

Диспетчер задач MS Windows позволяет просматривать общее использование памяти на вкладке **Быстродействие**.

Для запуска диспетчера задач можно использовать один из способов:

- Щелкнуть правой кнопкой мыши по свободному пространству на панели задач и выбрать из контекстного меню пункт Диспетчер задач (Task Manager)
- Нажать комбинацию клавиш <Ctrl>+<Shift>+<Esc>
- Нажать комбинацию клавиш <Ctrl>+<Alt>+<Del> и при необходимости нажать в открывшемся окне кнопку Диспетчер задач (Task Manager)
- Открыть окно Выполнить (Run) и ввести команду **taskmgn**

По умолчанию Windows создает файл подкачки в процессе загрузки ОС. Для установки размера файла подкачки нужно выполнить следующую последовательность действий. Выбрать значок **Мой компьютер** и выделить строку **Свойства**. Перейти на вкладку **Дополнительно** и нажать кнопку **Параметры** в рамке **Быстродействие**.

Инструмент, который может быть использован для определения рациональной интенсивности страничного обмена и оптимизации размера и размещения файла подкачки. - оснастка **Производительность**.

В **Примере решения** (Вариант 17) также дано более подробное руководство для выполнения этого практического задания.

Приступая к выполнению контрольных работ, рекомендуется ознакомиться со следующими методическими материалами:

1. Жадановская Н.П. Операционные системы. Базовый курс [Текст]: учебное пособие для студентов заочной формы обучения. – СПб.: СПбГТИ(ТУ), 2009. –xxx с.
2. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. Учебник для ВУЗов. - СПб.: ПИТЕР, 2007.- 544с.
3. Microsoft Windows XP Home Edition и Professional/ Русские версии/ Под общей редакцией А.Н. Чекмарева. – СПб.: БХВ-Петербург, 2006. - 640с.

В качестве руководства пользователя по операционной системе и Professional м.б. использовано любое популярное руководство пользователя.

## Контрольная работа 2: Вариант 1

### 1 Аналитическая часть

Сравните популярные стратегии замещения страниц. Оцените с точки зрения принципа оптимальности следующий нетрадиционный алгоритм выталкивания страниц «Истрепанная страница», при котором выталкивается страница, подвергшаяся наиболее интенсивным модификациям в системе.

### 2 Тестовые задания

**2.1** Защита памяти - это еще одна важная задача операционной системы, которая состоит в том, чтобы не позволить выполняемому процессу записывать или читать данные из памяти, назначенной другому .....

**2.2** Укажите основные недостатки распределения памяти фиксированными разделами:

- сложность реализации
- сложность защиты
- ограничение на число одновременно выполняющихся процессов
- фрагментация памяти

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 14 нс.
- размер страницы – 4 Кбайт.
- время работы процесса в системе – 144 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа №2: Вариант 2

### 1 Аналитический обзор

Поясните возможность организации мультипрограммирования при использовании прямой загрузки программ на выполнение, т.е когда процесс пользователя должен быть полностью размещен в основной памяти,

### 2 Тестовые задания

2.1 Память является важнейшим ресурсом, требующим тщательного управления со стороны мультипрограммной .....

2.2 При сегментно-страничной организации время от времени система выгружает ненужные страницы, освобождая оперативную память для новых:

- страниц
- блоков
- процессов
- разделов
- сегментов
- задач

### 3 Практические задания

3.1 Операционная система использует алгоритм замещения страниц **FIFO** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице **B2.2**:

Таблица **B2.2** - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 7 | 8 | 9 | 2 | 1 | 0 | 8 | 9 | 2 | 4  | 6  | 8  | 2  | 1  | 8  | 9  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 8 2 9 6
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

3.2 Используя штатные средства операционной системы MS Windows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 3

### 1 Аналитическая часть

Уточните, какие характеристики содержат таблица сегментов и таблица страниц при сегментно-страничной организации памяти.

### 2 Тестовые задания

2.1 Для уменьшения вероятности отсутствия страницы в памяти используются различные приемы, основным из которых является .....

2.2 Чтобы упростить механизм преобразования адресов при организации виртуальной памяти размер страницы в байтах специально выбирается равным:

- $2^{k+1}$
- $2^{k+3}$
- $2^{k-3}$
- $2^k$
- $2^{k-1}$

### 3 Практические задания

3.1 Операционная система использует алгоритм замещения страниц LRU в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.3:

Таблица В2.3 - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 8 | 6 | 0 | 5 | 4 | 8 | 6 | 2 | 3 | 9  | 1  | 8  | 6  | 0  | 3  | 6  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 7 1 8 6
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

3.2 Используя штатные средства операционной системы MS Windows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 4

### 1 Аналитическая часть

Сравните популярные стратегии замещения страниц. Оцените с точки зрения принципа оптимальности следующий нетрадиционный алгоритм выталкивания страниц «Глобальный LIFO», при котором выталкивается страница, самой последней поступившая в память.

### 2 Тестовые задания

**2.1** Для каждого процесса операционная система создает ..... — информационную структуру, содержащую записи обо всех виртуальных страницах процесса.

**2.2** Какой алгоритм выбора страницы для замещения близок к оптимальному:

- для замещения выбирается страница, дольше других находящаяся в памяти
- для замещения выбирается страница, выбранная случайным образом
- для замещения выбирается страница, к которой было меньше всего обращений за последний исследуемый промежуток времени

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 16 нс.
- размер страницы – 2 Кбайт.
- время работы процесса в системе – .86 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## 3.2. Контрольная работа 2: Вариант 5

### 1 Аналитическая часть

Перечислите стратегии управления страничной памятью.

### 2 Тестовые задания

**2.1** Механизм ..... памяти в большинстве универсальных операционных систем применяется ко всем сегментам пользовательской части виртуального адресного пространства процесса.

**2.2** При сегментной страничной организации в линейном виртуальном адресе искомым параметром виртуальной страницы (для работы страничного механизма) является: содержимое младших  $K$  разрядов - смещением, содержание старших разрядов:

- идентификатором
- приоритетом
- типом
- размером
- номером

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц **FIFO** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.5:

Таблица В2.5 - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 5 | 6 | 7 | 0 | 9 | 8 | 6 | 7 | 0 | 2  | 4  | 6  | 0  | 9  | 6  | 7  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 6 0 7 3
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 6

### 1 Аналитическая часть

Поясните, что понимается под стратегией рабочего множества, используемой при управлении виртуальной памятью

### 2 Тестовые задания

**2.1** Простейший способ управления оперативной памятью состоит в том, что память разбивается на несколько областей фиксированной величины, называемых .....

**2.2** К функциям операционной системы по управлению памятью относятся:

- выделение памяти потокам
- отслеживание свободной памяти
- выделение памяти процессам
- освобождение памяти после завершения процессов
- отслеживание занятой памяти
- освобождение памяти после завершения потоков

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц LRU в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице **В2.6**:

Таблица **В2.6** - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 4 | 2 | 6 | 1 | 0 | 4 | 2 | 8 | 9 | 5  | 7  | 4  | 2  | 6  | 9  | 2  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 3 7 4 2
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MS Windows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.



## Контрольная работа 2: Вариант 7

### 1 Аналитическая часть

Сравните популярные стратегии замещения страниц. Оцените с точки зрения принципа оптимальности следующий нетрадиционный алгоритм выталкивания страниц «Утомленная страница», при котором выталкивается страница, подвергшаяся наиболее интенсивным обращениям в системе.

### 2 Тестовые задания

2.1 Смежные в виртуальной памяти сегменты одного процесса могут занимать в оперативной памяти ..... участки.

2.2 При страничной организации виртуальной памяти время от времени операционная система выгружает ненужные страницы, освобождая оперативную память для новых:

- страниц
- блоков
- разделов
- сегментов
- процессов

### 3 Практические задания

3.1 Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определите, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 12 нс.
- размер страницы – 512 байт.
- время работы процесса в системе – 112 мс.

Оцените, как размер страницы влияет на производительность системы.

3.2 Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 8

### 1 Аналитическая часть

Укажите общие и отличительные признаки систем с чисто сегментной и комбинированной странично - сегментной организацией виртуальной памяти.

### 2 Тестовые задания

**2.1** Чем больше текущий размер страничного файла, тем ..... приложений может одновременно выполнять операционная система при фиксированном размере оперативной памяти.

**2.2** Какой из следующих методов распределения памяти может рассматриваться как частный случай виртуальной памяти:

- распределение фиксированными разделами;
- распределение динамическими разделами;
- страничное распределение;
- сегментное распределение;
- сегментно-страничное распределение
- распределение с использованием оверлеев.

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц **FIFO** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.8:

**Таблица В2.8** - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 0 | 1 | 2 | 5 | 4 | 3 | 1 | 2 | 5 | 6  | 9  | 1  | 5  | 4  | 1  | 2  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 1 5 2 7
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 9

### 1 Аналитическая часть

Дайте характеристику основных стратегий управления страничной памятью

### 2 Тестовые задания

**2.1** При сегментно-страничном распределении оперативной памяти виртуальное адресное пространство делится на сегменты, которые затем делятся на .....

**2.2** Уточните, какая стратегия управления виртуальной памятью определяет, какие конкретно данные необходимо удалять из памяти, чтобы поместить требуемые:

- загрузки
- замещения
- выборки
- размещения
- подкачки

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц LRU в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.9:

Таблица В2.9 - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 0 | 8 | 2 | 7 | 6 | 0 | 8 | 4 | 5 | 1  | 3  | 0  | 8  | 2  | 5  | 8  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 9 3 0 8
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 10

### 1 Аналитическая часть

Поясните, на что влияет размер страницы при виртуальной организации памяти и приведите аргументы за и против страницы малых размеров (страницы больших размеров).

### 2 Тестовые задания

**2.1** Подсистема виртуальной памяти представляет собой удобный механизм для решения задачи совместного доступа нескольких процессов к одному и тому же ..... памяти, который в этом случае называется разделяемой памятью

**2.2** В настоящее время все множество реализаций виртуальной памяти может быть представлено следующими классами:

- страничная
- оверлейная
- ассоциативная
- сегментная
- сегментно-страничная

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 24 нс.
- размер страницы – 1024 байт.
- время работы процесса в системе – 124 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 11

### 1 Аналитическая часть

Приведите сходства и различия между страничной и сегментной организацией виртуальной памяти.

### 2 Тестовые задания

**2.1** Все процессы, выполнившие системные вызовы с одним и тем же идентификатором, получают доступ к этому ..... и используют его по своему усмотрению, например в качестве буфера для обмена данными.

**2.2** Уточните, какая функция операционной системы по управления оперативной памятью характерна только для мульти задачных систем.

- выделение памяти по запросу
- освобождение памяти по завершению процесса
- защита памяти
- выделение памяти с упреждением
- учёт свободной и занятой памяти
- корректировка информации о состоянии памяти

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц **FIFO** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.11:

Таблица В2.11 - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 9 | 0 | 1 | 4 | 3 | 2 | 0 | 1 | 4 | 6  | 8  | 0  | 4  | 3  | 0  | 1  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 0 4 1 7
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 12

### 1 Аналитическая часть

Уточните структуры таблиц сегментов и страниц для организации виртуальной памяти и поясните возможность организации защиты в подобной системе.

### 2 Тестовые задания

**2.1** При работе с разделяемыми ..... памяти операционная система должна выполнять некоторые функции, общие для любых разделяемых между процессами ресурсов — файлов, семафоров и т. п.

**2.2** Совокупность виртуальных адресов процесса называется виртуальным адресным пространством. Виртуальные адреса являются результатом работы:

- пользователя
- транслятора
- компоновщика
- ассемблера
- инсталлятора

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц LRU в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.12:

Таблица В2.12 - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 2 | 0 | 4 | 9 | 8 | 2 | 0 | 6 | 7 | 3  | 5  | 2  | 0  | 4  | 7  | 0  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 1 5 2 0
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 13

### 1 Аналитическая часть

Уточните структуры таблиц для организации виртуальной памяти и поясните возможность организации защиты в подобной системе.

### 2 Тестовые задания

**2.1.** В любом случае операционная система обычно следит за корректностью использования процессом виртуальных адресов — процессу не разрешается оперировать с виртуальным адресом, выходящим за пределы назначенных ему .....

**2.2** Для каждого процесса операционная система создает таблицу страниц, информация из которой используется для:

- преобразования виртуального адреса в физический
- ускорения работы процесса
- решения вопроса о необходимости перемещения той или иной страницы между памятью и диском,
- учета смежных страниц в виртуальной памяти и контроля на их смежность в физической памяти
- разбиения программы на части, называемые оверлеями.

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 11 нс.
- размер страницы – 1 Кбайт.
- время работы процесса в системе – 111 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 14

### 1 Аналитическая часть

Перечислите основные функции операционной системы по управлению основной памятью в мультипрограммной системе

### 2 Тестовые задания

**2.1** Чем меньше текущий размер страничного файла, тем ..... приложений может одновременно выполнять операционная система при фиксированном размере оперативной памяти.

**2.2** Определите, какое из высказываний истинно:

- виртуальное адресное пространство всегда равно 2 Мбайт
- виртуальное адресное пространство всегда совпадает по размеру с размером физической памяти, имеющейся в компьютере
- виртуальное адресное пространство процесса определяется разрядностью адресной шины компьютера
- виртуальное адресное пространство всегда равно двум размерам физической памяти

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц **FIFO** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.14:

**Таблица В2.14** - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 2 | 3 | 4 | 7 | 6 | 5 | 3 | 4 | 7 | 8  | 1  | 3  | 7  | 6  | 3  | 4  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 3 7 4 9
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.



## Контрольная работа 2: Вариант 15

### 1 Аналитическая часть

Перечислите основные классы реализаций виртуальной памяти

### 2 Тестовые задания

**2.1** В не вытесняемой области ..... памяти размещаются модули ОС, требующие быстрой реакции или постоянного присутствия в оперативной памяти.

**2.2** Свопинг представляет собой способ совместного использования оперативной памяти и диска. Уточните, что из ниже перечисленного верно для свопинга:

- на диск выгружается неиспользуемая в настоящий момент часть процесса
- на диск выгружается неиспользуемая в настоящий момент часть потока
- на диск выгружаются неиспользуемые процессом данные
- на диск выгружается не активный процесс
- на диск выгружаются переменные окружения системы
- на диск выгружаются неиспользуемые процессом данные

### 3 Практические задания

**3.1** Операционная система использует алгоритм замещения страниц **LRU** в системе с четырьмя страничными блоками и восьмью страницами. Последовательность обращений к страницам приведена в таблице В2.15:

**Таблица В2.15** - Последовательность обращений

|           |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |
|-----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| обращение | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| страница  | 6 | 4 | 8 | 3 | 2 | 6 | 4 | 0 | 1 | 7  | 9  | 6  | 4  | 8  | 1  | 4  |

Требуется определить:

- сколько страничных прерываний произойдет в системе при условии, что четыре страничных блока изначально заняты страницами: 5 9 6 4
- сколько страничных прерываний произойдет в системе при увеличении количества страничных блоков на 1
- наличие (или отсутствие) более оптимального алгоритма замещения.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 16

### 1 Аналитическая часть

Покажите основные преимущества и недостатки виртуальной организации памяти.

### 2 Тестовые задания

**2.1** Процесс при виртуальной организации памяти находится в состоянии ....., если он больше времени занимается подкачкой страниц, нежели выполнением.

**2.2** Уточните, какой алгоритм выбора страницы для замещения близок к оптимальному:

- для замещения выбирается страница, дольше других находящаяся в памяти
- для замещения выбирается страница, выбранная случайным образом
- для замещения выбирается страница, к которой было меньше всего обращений за последний исследуемый промежуток времени
- для замещения выбирается страница, подвергшаяся наиболее интенсивным модификациям в системе
- для замещения выбирается страница, поступившая первой

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 18 нс.
- размер страницы – 2048 байт.
- время работы процесса в системе – 148 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 17 - Примеры решения

### 1 Аналитическая часть

Поясните, что понимается под явлением пробуксовка (Thrashing), являющимся исключительной ситуацией при работе с виртуальной памятью.

### 2 Тестовые задания

**2.1** Диапазон возможных адресов виртуального пространства у всех процессов является одним и тем же. Однако, в разных операционных системах используются разные способы структуризации ..... адресного пространства.

**2.2** Простейший способ управления оперативной памятью состоит в том, что память разбивается на несколько областей, называемых разделами. Уточните, какой процесс обязательно должен выполняться в системе организации памяти с перемещаемыми разделами:

- перемещение
- блокирование
- сжатие
- свопинг (swapping)
- пэйджинг (paging)

### 3 Практические задания

**3.1** Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.

При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.

Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:

- копирование 1 слова требует – 18 нс.
- размер страницы – 2048 байт.
- время работы процесса в системе – 148 мс.

Оцените, как размер страницы влияет на производительность системы.

**3.2** Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение.

Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью.

Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты.

Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

## Контрольная работа 2: Вариант 17 - Примеры решения

### 1 Аналитический обзор

Напишите ответ на поставленный вопрос:

Поясните, что понимается под явлением пробуксовка (Thrashing), являющимся исключительной ситуацией при работе с виртуальной памятью.

Программное обеспечение подсистемы управления виртуальной памятью операционной системы -ОС - связано с реализацией следующих стратегий:

- **Стратегия выборки (fetch policy)** - в какой момент следует переписать страницу из вторичной памяти в первичную.

Существует два основных варианта выборки: по запросу и с упреждением. Алгоритм выборки по запросу вступает в действие в тот момент, когда процесс обращается к отсутствующей странице. Его реализация заключается в загрузке страницы с диска в свободную физическую страницу и коррекции соответствующей записи таблицы страниц. Алгоритм выборки с упреждением осуществляет опережающее чтение, при котором кроме страницы, вызвавшей исключительную ситуацию, в память также загружается несколько страниц, окружающих ее. Такой алгоритм призван уменьшить накладные расходы, связанные с большим количеством исключительных ситуаций.

- **Стратегия размещения (placement policy)** - в какой участок первичной памяти поместить поступающую страницу.

В случае систем с сегментной организацией необходима стратегия, аналогичная стратегии с динамическим распределением. В системах со страничной организацией все проще - в любой свободный страничный кадр.

- **Стратегия замещения (replacement policy)** - какую страницу нужно вытолкнуть во внешнюю память, чтобы освободить место в оперативной памяти. Разумная стратегия замещения позволяет хранить в памяти самую необходимую информацию и тем самым снизить частоту страничных нарушений – прерываний по отсутствию нужной страницы в памяти.

Итак, наиболее ответственным действием менеджера памяти является выделение кадра оперативной памяти для размещения в нем виртуальной страницы, находящейся во внешней памяти. В этом случае ОС в соответствии с заложенными в нее критериями должна:

- найти некоторую занятую страницу основной памяти;
- переместить ее содержимое во внешнюю память;
- переместить в этот страничный кадр нужную виртуальную страницу из внешней памяти;
- модифицировать соответствующие таблицы страниц;
- продолжить выполнение процесса, потребовавшего эту виртуальную страницу.

**Т.о. при замещении приходится дважды передавать страницу между основной и вторичной памятью.**



Как видно на рисунке В2.17.1.а система с тремя страничными кадрами (9 раз имеет место page faults) оказывается более производительной, чем с четырьмя страничными кадрами (10 раз имеет место page faults), для строки обращений к памяти 0 1 2 3 0 1 4 0 1 2 3 4 при выборе стратегии FIFO.

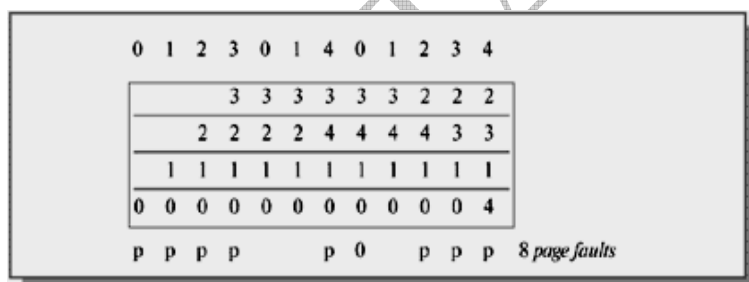
▪ **Оптимальный алгоритм (OPT)**

Аномалия Билэди иллюстрирует сложность ОС, где интуитивный подход не всегда приемлем. Одним из последствий открытия аномалии Билэди стал поиск оптимального алгоритма, который при заданной строке обращений имел бы минимальную частоту page faults среди других алгоритмов.

Такой алгоритм был найден – необходимо замещать страницу, которая не будет использоваться в течение самого длительного периода времени. Этот алгоритм легко описать, но реализовать практически невозможно. ОС не знает, к какой странице будет следующее обращение.

▪ **Алгоритм LRU - выталкивание дольше всего не использовавшейся страницы**

Одним из приближений к алгоритму OPT является алгоритм замещения страницы, которая не использовалась в течение самого долгого времени. Работа алгоритма проиллюстрирована на рисунке В2.17.2. Сравнивая рисунки В2.17.1 В2.17.2, можно увидеть, что использование алгоритма LRU позволяет сократить количество page faults.



**Рисунок В2.17.2 - Пример работы алгоритма LRU**

LRU - хороший, но труднореализуемый алгоритм. Необходимо иметь связанный список всех страниц в памяти, в начале которого будут храниться недавно использованные страницы. Причем этот список должен обновляться при каждом обращении к памяти. Много времени нужно также и на поиск страниц в таком списке.

Как оптимальный алгоритм, так и LRU не страдают от аномалии Билэди.

▪ **Алгоритм NFU - выталкивание редко используемой страницы**

Поскольку большинство современных процессоров не предоставляют соответствующей аппаратной поддержки для реализации алгоритма LRU, хотелось бы иметь алгоритм, достаточно близкий к LRU, но не требующий специальной поддержки. Программная реализация алгоритма, близкого к LRU - алгоритм NFU (Not Frequently Used). Для него требуются программные счетчики, по одному на каждую страницу, которые сначала равны нулю. При каждом прерывании по времени ОС сканирует все страницы в памяти и у каждой страницы с установленным флагом обращения увеличивает на единицу

значение счетчика, а флаг обращения сбрасывает. Т.о, освобождается страница с наименьшим значением счетчика, как страница, к которой реже всего обращались. Довольно устойчивым недостатком алгоритма является длительность процесса сканирования таблиц страниц.

▪ **Working Set Page Replacement - стратегия рабочего множества**

В современных ОС популярен алгоритм, основанный на управлении количеством страниц, выделенным процессу - модель рабочего множества. Рассмотрим более общий подход, базирующийся на концепции рабочего множества: необходимо решить, что делать, если в распоряжении процесса имеется недостаточное число кадров, т.е. нужно ли его приостановить с освобождением всех кадров, и что следует понимать под достаточным количеством кадров.

Теоретически возможно уменьшить число кадров процесса до минимума, однако существует какое-то число активно используемых страниц, без которого процесс часто генерирует page faults.

**Высокая частота генерации page faults - страничных нарушений называется thrashing, иногда употребляется русский термин «пробуксовка» (см. рисунок В2.17.3).**



**Рисунок В2.17.3 - Зависимость частоты page faults от количества кадров**

Процесс находится в состоянии **thrashing**, если при его работе больше времени уходит на подкачку страниц, нежели на выполнение команд. Такого рода критическая ситуация возникает вне зависимости от конкретных алгоритмов замещения.

Результатом **thrashing** является резкое снижение производительности вычислительной системы. Например, при глобальном алгоритме замещения процесс, которому не хватает кадров, начинает отбирать кадры у других процессов, которые в свою очередь начинают заниматься тем же. В результате все процессы попадают в очередь запросов к устройству вторичной памяти. Загрузка процессора резко снижается. ОС реагирует на это увеличением степени мультипрограммирования, что приводит к еще большему **thrashing** и дальнейшему снижению загрузки процессора.

Эффект **thrashing**, возникающий при использовании глобальных алгоритмов, может быть ограничен за счет применения локальных алгоритмов

замещения, при которых если даже один из процессов попал в **thrashing**, это не сказывается на других процессах. Однако он много времени проводит в очереди к устройству выгрузки, затрудняя подкачку страниц остальных процессов.

Итак, **thrashing** - это высокая частота страничных нарушений, которую надо контролировать. Очевидно, когда она высока, процесс нуждается в дополнительных кадрах. Можно, устанавливая желаемую частоту page faults, регулировать размер процесса, добавляя или отнимая у него кадры. Может оказаться целесообразным выгрузить процесс целиком. Освободившиеся кадры выделяются другим процессам с высокой частотой page faults.

Для предотвращения **thrashing** требуется выделять процессу столько кадров, сколько ему нужно, поэтому необходимо попытаться выяснить, как много кадров процесс реально использует. Для решения этой задачи и используется модель рабочего множества.

При рассмотрении поведения реальных процессов можно заметить, что процессы начинают работать, не имея в памяти необходимых страниц и в результате при выполнении первой инструкции возникает page fault, требующий подкачки нужной страницы, следующий page fault происходит при локализации глобальных переменных и еще один - при выделении памяти для стека. После того как процесс собрал большую часть необходимых ему страниц, page faults возникают редко.

Т.о., существует набор страниц, который позволяет процессу в течение некоторого периода времени производительно работать, избегая большого количества page faults. **Этот набор страниц называется рабочим множеством (working set) процесса.** Число страниц в нем относительно невелико.

Важное свойство рабочего множества - его размер. ОС должна выделить процессу число кадров, в которое поместилось бы его рабочее множество.

Если кадры еще остались, то может быть инициирован другой процесс. Если рабочие множества процессов не помещаются в память и начинается **thrashing**, то один из процессов можно выгрузить на диск.

Т.о., решение о размещении процессов в памяти должно базироваться на размере его рабочего множества.

Один из способов реализации данного подхода заключается в том, что для впервые иницируемых процессов это решение м.б. принято эвристически. Во время работы процесса ОС должна уметь определять: расширяет процесс свое рабочее множество или перемещается на новое рабочее множество.

Другой способ реализации данного подхода может быть основан на отслеживании количества страничных нарушений, вызываемых процессом. Если процесс часто генерирует page faults и память не слишком заполнена, то система может увеличить число выделенных ему кадров. Если же процесс не вызывает page faults в течение некоторого времени и уровень генерации ниже какого-то порога, то число кадров процесса может быть урезано. Этот способ регулирует лишь размер множества страниц, принадлежащих процессу, и должен быть дополнен какой-либо стратегией замещения страниц. Несмотря на то что система при этом может «пробуксовывать» в моменты перехода от



одного рабочего множества к другому, предлагаемое решение в состоянии обеспечить наилучшую производительность для каждого процесса, не требуя никакой дополнительной настройки системы.

Подсистема виртуальной памяти работает эффективно при наличии резерва свободных страничных кадров. Алгоритмы, обеспечивающие поддержку системы в состоянии отсутствия **thrashing**, реализованы в составе фоновых процессов (их часто называют демонами или сервисами), которые периодически «просыпаются» и инспектируют состояние памяти. Если свободных кадров оказывается мало, они могут сменить стратегию замещения. Их задача - поддерживать систему в состоянии наилучшей производительности.

Примером такого рода процесса может быть фоновый процесс - сборщик страниц, реализующий облегченный вариант алгоритма откачки, основанный на использовании рабочего набора. Он начинает активно работать, когда количество страниц в списке свободных страниц достигает установленного нижнего порога. Но если возникает требование страницы в условиях, когда список свободных страниц пуст, то начинает работать механизм свопинга, поскольку простое отнятие страницы у любого процесса потенциально вело бы к ситуации **thrashing**, и разрушало бы рабочий набор некоторого процесса.

Такой демон применяется во многих клонах ОС Unix.

В ОС Windows 2000/XP такую роль играет менеджер балансного набора (Working set manager), который вызывается раз в секунду или тогда, когда размер свободной памяти опускается ниже определенного предела, и отвечает за суммарную политику управления памятью и поддержку рабочих множеств.

## 2 Тестовые задания

### 2.1 Вставьте пропущенное слово

**Диапазон возможных адресов виртуального пространства у всех процессов является одним и тем же. Однако, в разных операционных системах используются разные способы структуризации ..... адресного пространства.**

Правильным ответом на данный вопрос является: **виртуального адресного процесса.**

Память является важнейшим ресурсом, требующим тщательного управления со стороны мультипрограммной **операционной системы - ОС.** Особая роль памяти объясняется тем, что процессор может выполнять инструкции только в том случае, если они находятся в памяти.

Память распределяется как между модулями прикладных программ, так и между модулями самой ОС.

В ранних ОС управление памятью сводилось просто к загрузке программы и ее данных из некоторого внешнего накопителя (перфоленты, магнитной ленты или магнитного диска) в память.

С появлением мультипрограммирования перед ОС были поставлены новые задачи, связанные с распределением имеющейся памяти между несколькими одновременно выполняющимися программами.

Функциями ОС по управлению памятью в мультипрограммной системе являются:

- отслеживание свободной и занятой памяти;
- выделение памяти процессам и освобождение памяти по завершении процессов;
- вытеснение кодов и данных процессов из основной памяти на диск, когда размеры основной памяти не достаточны для размещения в ней всех процессов, и возвращение их в основную память, когда в ней освобождается место;
- настройка адресов программы на конкретную область физической памяти.

Помимо первоначального выделения памяти процессам при их создании ОС должна также заниматься динамическим распределением памяти, то есть выполнять запросы приложений на выделение им дополнительной памяти во время выполнения. После того как приложение перестает нуждаться в дополнительной памяти, оно может вернуть ее системе.

Для идентификации переменных и команд на разных этапах жизненного цикла программы используются символьные имена (метки), виртуальные адреса и физические адреса:

- Символьные имена присваивает пользователь при написании программы на алгоритмическом языке или ассемблере.
- Виртуальные адреса, называемые иногда математическими, или логическими адресами, вырабатывает транслятор, переводящий программу на машинный язык. Поскольку во время трансляции в общем случае не известно, в какое место оперативной памяти будет загружена программа, то транслятор присваивает переменным и командам виртуальные (условные) адреса, обычно считая по умолчанию, что начальным адресом программы будет нулевой адрес.
- Физические адреса соответствуют номерам ячеек оперативной памяти, где в действительности расположены' или будут расположены переменные и команды.

Совокупность виртуальных адресов процесса называется виртуальным адресным пространством. Диапазон возможных адресов виртуального пространства у всех процессов является одним и тем же. Например, при использовании 32-разрядных виртуальных адресов этот диапазон задается границами 0000000016и FFFFFFFF16. Тем не менее каждый процесс имеет собственное виртуальное адресное пространство — транслятор присваивает виртуальные адреса переменным и кодам каждой программе независимо.

В разных ОС используются разные способы структуризации виртуального адресного пространства.

В одних ОС виртуальное адресное пространство процесса подобно физической памяти представлено в виде непрерывной линейной последовательности виртуальных адресов. Таковую структуру адресного пространства называют также плоской (flat). При этом виртуальным адресом является единственное число, представляющее собой смещение относительно начала (обычно это значение 000...000) виртуального адресного пространства. Адрес такого типа называют линейным виртуальным адресом.

В других ОС виртуальное адресное пространство делится на блоки. В этом случае помимо линейного адреса может быть использован виртуальный адрес, представляющий собой пору чисел  $(i, t)$ , где  $i$  определяет блок, а  $t$  — смещение внутри блока. Существуют и более сложные способы структуризации виртуального адресного пространства, когда виртуальный адрес образуется тремя или даже более числами.

Итак, ключевой проблемой виртуальной памяти, возникающей в результате многократного изменения местоположения в оперативной памяти образов процессов или их частей, является преобразование виртуальных адресов в физические. Решение этой проблемы, в свою очередь, зависит от того, какой способ структуризации виртуального адресного пространства принят в данной системе управления памятью.

В настоящее время все множество реализаций виртуальной памяти может быть представлено тремя классами.

- **Страничная виртуальная память** организует перемещение данных между памятью и диском страницами — частями виртуального адресного пространства, фиксированного и сравнительно небольшого размера.

- **Сегментная виртуальная память** предусматривает перемещение данных сегментами — частями виртуального адресного пространства произвольного размера, полученными с учетом смыслового значения данных,

- **Сегментно-страничная виртуальная память** использует двухуровневое деление: виртуальное адресное пространство делится на сегменты, а затем сегменты делятся на страницы. Единицей перемещения данных здесь является страница. Этот способ управления памятью объединяет в себе элементы обоих предыдущих подходов.

Для временного хранения сегментов и страниц на диске отводится либо специальная область, либо специальный файл, которые во многих ОС по традиции продолжают называть областью, или файлом свопинга, хотя перемещение информации между оперативной памятью и диском осуществляется уже не в форме полного замещения одного процесса другим, а частями. Другое популярное название этой области — страничный файл (page file, или paging file).

Текущий размер страничного файла является важным параметром, оказывающим влияние на возможности ОС: чем больше страничный файл, тем больше приложений может одновременно выполнять ОС (при фиксированном размере оперативной памяти). Однако необходимо понимать, что увеличение числа одновременно работающих приложений за счет увеличения размера страничного файла замедляет их работу, так как значительная часть времени при этом тратится на перекачку кодов и данных из оперативной памяти на диск и обратно.

Размер страничного файла в современных ОС является настраиваемым параметром, который выбирается администратором системы для достижения компромисса между уровнем мультипрограммирования и быстродействием системы.

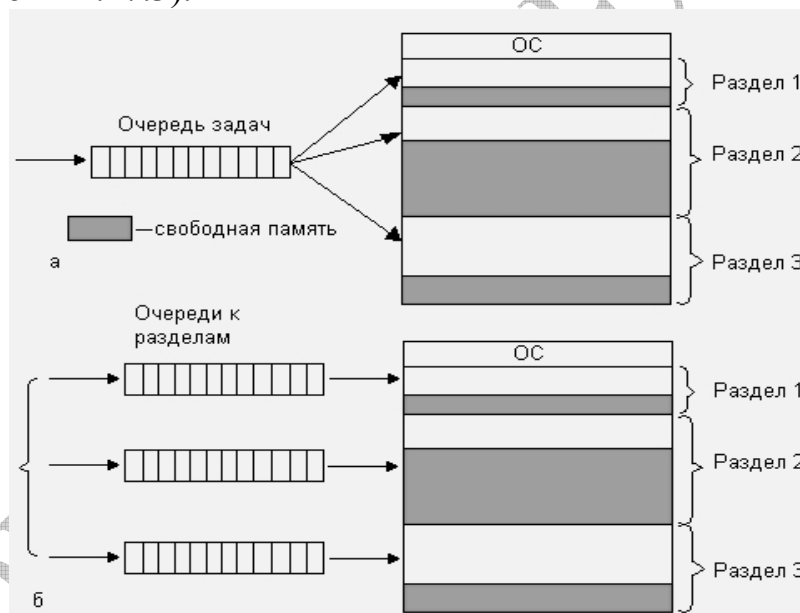
**2.2 Выберите все правильные ответы на предложенное тестовое задание:**

**Простейший способ управления оперативной памятью состоит в том, что память разбивается на несколько областей, называемых разделами. Уточните, какой процесс обязательно должен выполняться в системе организации памяти с перемещаемыми разделами:**

- перемещение
- блокирование
- сжатие
- свопинг (swapping)
- пэйджинг (paging)

Простейший способ управления оперативной памятью состоит в том, что память разбивается на несколько областей фиксированной величины, называемых разделами. После этого границы разделов не изменяются.

Очередной новый процесс, поступивший на выполнение, помещается либо в общую очередь (рисунок В2.17.3, а), либо в очередь к некоторому разделу (рисунок В2.17.3).



**Рисунок В2.17.3 - Распределение памяти фиксированными разделами: с общей очередью (а), с отдельными очередями (б)**

Подсистема управления памятью в этом случае выполняет следующие задачи:

- Сравнивает объем памяти, требуемый для вновь поступившего процесса, с размерами свободных разделов и выбирает подходящий раздел;
- Осуществляет загрузку программы в один из разделов и настройку адресов.

При очевидном преимуществе — простоте реализации, данный метод имеет существенный недостаток — жесткость. Так как в каждом разделе может выполняться только один процесс, то уровень мультипрограммирования

заранее ограничен числом разделов. Независимо от размера программы она будет занимать весь раздел. Так, например, в системе с тремя разделами невозможно выполнять одновременно более трех процессов, даже если им требуется совсем мало памяти. С другой стороны, разбиение памяти на разделы не позволяет выполнять процессы, программы которых не помещаются ни в один из разделов, но для которых было бы достаточно памяти нескольких разделов.

Такой способ управления памятью применялся в ранних мультипрограммных ОС. Однако и сейчас метод распределения памяти фиксированными разделами находит применение в системах реального времени, в основном благодаря небольшим затратам на реализацию. Детерминированность вычислительного процесса систем реального времени (заранее известен набор выполняемых задач, их требования к памяти, а иногда и моменты запуска) компенсирует недостаточную гибкость данного способа управления памятью.

Как улучшение вышеизложенного способа предлагается распределение памяти динамическими разделами. В этом случае память машины не делится заранее на разделы. Сначала вся память, отводимая для приложений, свободна. Каждому вновь поступающему на выполнение приложению на этапе создания процесса выделяется вся необходимая ему память (если достаточный объем памяти отсутствует, то приложение не принимается на выполнение и процесс для него не создается). После завершения процесса память освобождается, и на это место может быть загружен другой процесс. Таким образом, в произвольный момент времени оперативная память представляет собой случайную последовательность занятых и свободных участков (разделов) произвольного размера.

На рисунке В2.17.4 показано состояние памяти в различные моменты времени при использовании динамического распределения. Так, в момент  $t_0$  в памяти находится только ОС, а к моменту  $t_1$  память разделена между 5 процессами, причем процесс П4, завершаясь, покидает память. На освободившееся от процесса П4 место загружается процесс П6, поступивший в момент  $t_3$ .

Функции ОС, предназначенные для реализации данного метода управления памятью, перечислены ниже.

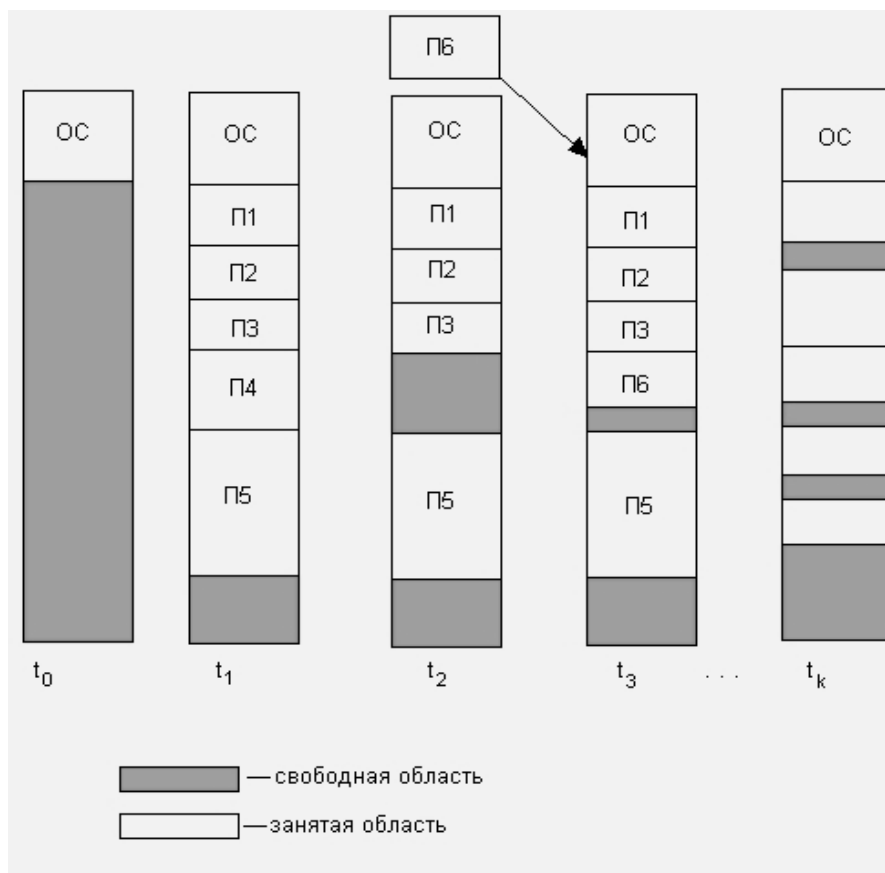
- Ведение таблиц свободных и занятых областей, в которых указываются начальные адреса и размеры участков памяти.

- При создании нового процесса — анализ требований к памяти, просмотр таблицы свободных областей и выбор раздела, размер которого достаточен для размещения кодов и данных нового процесса. Выбор раздела может осуществляться по разным правилам, например: «первый попавшийся раздел достаточного размера», «раздел, имеющий наименьший достаточный размер» или «раздел, имеющий наибольший достаточный размер».

- Загрузка программы в выделенный ей раздел и корректировка таблиц свободных и занятых областей. Данный способ предполагает, что программный

код не перемещается во время выполнения, а значит, настройка адресов может быть проведена единовременно во время загрузки.

- После завершения процесса корректировка таблиц свободных и занятых областей.



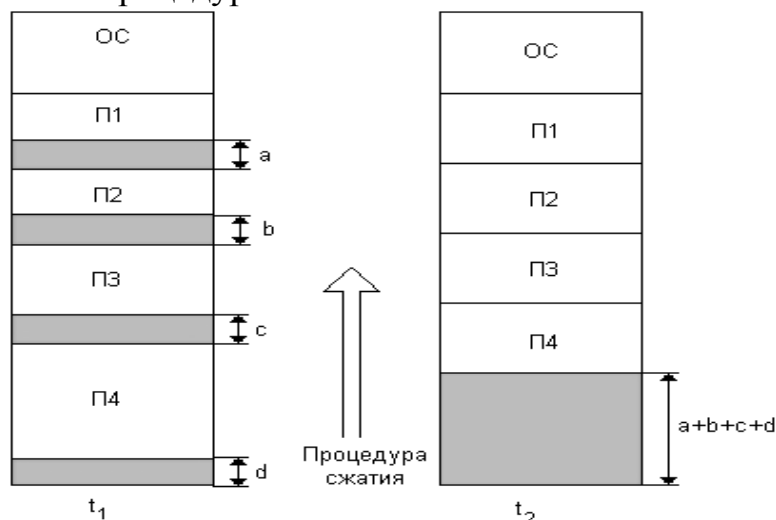
**Рисунок В2.17.4** - Распределение памяти динамическими разделами

По сравнению с методом распределения памяти фиксированными разделами данный метод обладает гораздо большей гибкостью, но ему присущ очень серьезный недостаток — фрагментация памяти.

Фрагментация — это наличие большого числа несмежных участков свободной памяти очень маленького размера (фрагментов). Настолько маленького, что ни одна из вновь поступающих программ не может поместиться ни в одном из участков, хотя суммарный объем фрагментов может составить значительную величину, намного превышающую требуемый объем памяти.

Одним из методов борьбы с фрагментацией является перемещение всех занятых участков в сторону старших или младших адресов, так, чтобы вся свободная память образовала единую свободную область (рисунок В2.17.5). В дополнение к функциям, которые выполняет ОС при распределении памяти динамическими разделами в данном случае она должна еще время от времени копировать содержимое разделов из одного места памяти в другое, корректируя таблицы свободных и занятых областей. Эта процедура называется **сжатием**. Сжатие может выполняться либо при каждом завершении процесса, либо только тогда, когда для вновь создаваемого процесса нет свободного раздела

достаточного размера. В первом случае требуется меньше вычислительной работы при корректировке таблиц свободных и занятых областей, а во втором — реже выполняется процедура сжатия.



**Рисунок В2.17.5** - Распределение памяти перемещаемыми разделами

Так как программы перемещаются по оперативной памяти в ходе своего выполнения, то в данном случае невозможно выполнить настройку адресов с помощью перемещающего загрузчика. Здесь более подходящим оказывается динамическое преобразование адресов.

Хотя процедура сжатия и приводит к более эффективному использованию памяти, она может потребовать значительного времени, что часто перевешивает преимущества данного метода.

Концепция сжатия применяется и при использовании других методов распределения памяти, когда отдельному процессу выделяется не одна сплошная область памяти, а несколько несмежных участков памяти произвольного размера (сегментов).

### 3 Практические задания

#### 3.1 Предложите решение теоретической задачи

**Компьютер имеет 32 – разрядное адресное пространство. Организация памяти – страничная. Таблица страниц полностью поддерживается аппаратно и на запись в ней отводится одно 32 разрядное слово.**

**При запуске пользовательского процесса таблица страниц копируется из памяти в аппаратуру. Время работы процесса включает время загрузки таблицы страниц.**

**Определить, какая доля времени процессора жертвуется на загрузку таблицы страниц считая что:**

- копирование 1 слова требует 10 нс.
- размер страницы – 8 Кбайт.
- время работы процесса в системе - 100 мс.

**Оцените, как размер страницы влияет на производительность системы.**

### Решение задачи предлагается следующее:

При известном времени записи слова в аппаратуру для определения времени копирования страниц необходимо знать число слов в таблице.

Оно должно быть равно числу виртуальных страниц в виртуальном адресном пространстве.

Размер виртуального адресного пространства **Ева** определяется разрядностью адреса, которая равна 32. Следовательно:

$$\text{Ева} = 2^{32} \text{ байт}$$

При заданном размере страницы  $\text{Естр} = 8$  кбайт число виртуальных страниц  $\text{Nстр.}$  равно:

$$\text{Nстр} = \text{Ева} / \text{Естр} = 2^{32} / 2^{13} = 2^{19}$$

Теперь можно найти время копирования таблицы страниц в аппаратуру –  $\text{Ткоп\_табл}$  с учетом времени копирования одного слова –  $\text{Ткоп\_слова}$

$$\text{Ткоп\_табл} = \text{Ткоп\_слова} * \text{Nстр} = 10 * 2^{19} = 5242880 \text{ (нс)} = 5,24 \text{ (мск)}$$

Так как время работы процесса  $\text{Тпроцесса}$  равно 100 мс, то доля времени процессора -  $\Delta$ , затрачиваемая на загрузку таблицы страниц будет равна

$$\Delta = (\text{Ткоп\_табл} / \text{Тпроцесса}) * 100\% = 5,24 / 100 * 100\% = 5,24\%$$

Очевидно, что при уменьшении размера страницы, например в 2 раза, увеличивается количество создаваемых и используемых страниц:

$$\text{Nстр} = \text{Ева} / \text{Естр} = 2^{32} / 2^{13} = 2^{19} * 2$$

Естественно, время копирования таблицы страниц в аппаратуру –  $\text{Ткоп\_табл}$  с учетом времени копирования одного слова –  $\text{Ткоп\_слова}$  также возрастет в 2 раза:

$$\text{Ткоп\_табл} = \text{Ткоп\_слова} * \text{Nстр} = 10 * 2^{19} * 2 = 5242880 * 2 \text{ (нс)} = 10,48 \text{ (мск)}$$

И увеличится доля времени процессора, затрачиваемая на загрузку таблицы страниц:

$$\Delta = 5,24\% * 2 = 10,48\%$$

Т.о. при уменьшении размера страниц доля времени процессора, затрачиваемая на загрузку таблицы страниц будет уменьшаться. И наоборот, при увеличении размера страниц доля времени процессора, затрачиваемая на загрузку таблицы страниц будет уменьшаться.

Однако, есть еще один важный фактор, связанный с размером страницы и влияющий на производительность системы - потери основной памяти процесса из-за наличия внутренней фрагментации при блочном распределении основной памяти, при котором процессу выделяется целое число страниц. Эти потери возрастают при увеличении размера страниц и уменьшаются при уменьшении размера страниц.



Конечно же размер страницы повлияет на уровень мультизадачности.

Размер страницы – всегда компромиссная задача В системах семейства MS Windows размер страницы равен 4 К.

### 3.2 Выполните предложенное практическое задание

Используя штатные средства операционной системы MSWindows XP определите для своего компьютера объем установленной физической памяти, объем виртуальной памяти, величину файла подкачки и его размещение. Поясните назначение тех счетчиков оснастки производительность, которые касаются деятельности подсистемы управления памятью. Исследуйте влияние размера файла подкачки на общую производительность вашей системы. Объясните полученные результаты. Предложите способ, позволяющий оптимизировать использование файла подкачки для повышения производительности.

В качестве тестовой системы для выполнения практического задания использовалась: операционная система – ОС- MS Windows XP PRO SP3.

Диспетчер задач MS Windows позволяет просматривать общее использование памяти на вкладке Быстродействие (см. рисунок В217.6). Здесь отображается информация в трех разделах: Выделение памяти, физическая память и Память ядра.

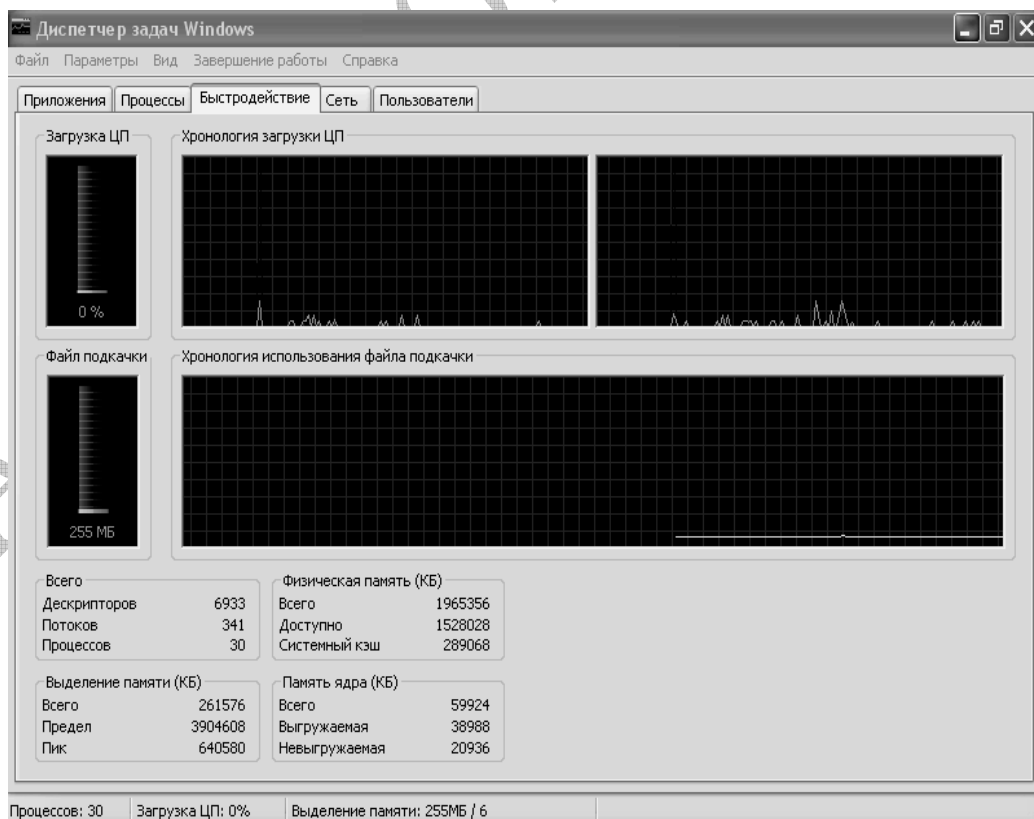


Рисунок В2.17.6 – Окно диспетчера задач MS Windows

В первом разделе содержится три статических параметра виртуальной памяти:

- Параметр **Всего** – это общий объем виртуальной памяти, используемой как приложениями, так и ОС.
- Параметр **Предел** – это объем доступной виртуальной памяти.
- Параметр **Пик** – наибольший объем памяти, использованный с момента последней загрузки.

В разделе **Физическая память** содержатся параметры, несущие информацию о текущем состоянии физической памяти машины. Эта статистика не имеет никакого отношения к файлу подкачки, следовательно, может являться хорошим индикатором ситуации, когда его увеличение не даст эффекта:

- Параметр **Всего** - это объем памяти, обнаруженный **ОС** на компьютере.
- Параметр **Доступно** отражает память, доступную для использования процессами. Эта величина не включает в себя память, доступную приложениям за счет файла подкачки. Каждое приложение требует определенный объем физической памяти и не может использовать только ресурсы файла подкачки.
- Параметр **Системный кэш** сообщает объем, доступный кэш-памяти системы. Это объем физической памяти, оставленный **ОС** после удовлетворения своих потребностей.

В разделе **Память ядра** отображается информация о потребностях компонентов ОС, обладающих наивысшим приоритетом. Эти компоненты : обычно работают с сервисом низкого уровня, типа прямого доступа к жесткому диску. Параметры **Память ядра** отображают потребности ключевых служб ОС:

- Параметр **Всего** - это объем виртуальной памяти, необходимой ОС.
- Параметр **Выгружаемая** - несет информацию об общем объеме памяти, использованной системой за счет файла подкачки.
- Параметр **Невыгружаемая** - объем физической памяти, потребляемой ОС.

Необходимо помнить, что эти параметры относятся лишь к привилегированным службам, а не ко всему сервису системы в целом. Многие компоненты **ОС** работают как приложения. В большинстве случаев параметры **Память ядра** должны оставаться без изменений, если не меняется что-либо в ядре **ОС** (например, устанавливается новое устройство в компьютер). Глобальные изменения в этом разделе обычно являются сигналом возможного возникновения проблем.

**ОС** семейства Windows XP в Службных программах содержит программу **Сведения о системе**, с помощью которой также можно получить сведения об основных характеристиках организации памяти в компьютере (см. рисунок В217.7).

В частности, здесь можно узнать полный объем установленной в компьютере физической памяти, общий объем виртуальной памяти и доступной (свободной) в данный момент времени виртуальной памяти, размещение и объем файла подкачки.

**Файл подкачки** - это область жесткого диска, используемая Windows для хранения данных оперативной памяти.

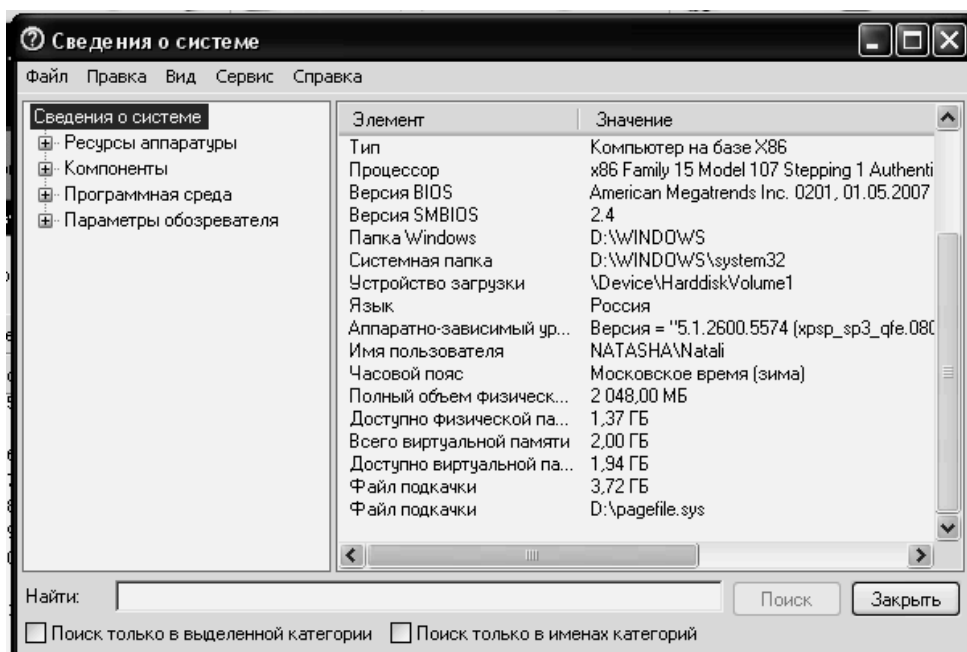


Рисунок В2.17.7 – Сведения о системе

Он создает иллюзию, что система располагает большим объемом оперативной памяти, чем это есть на самом деле. Единой стратегии работы с файлом подкачки не существует. Многое определяется назначением и настройкой компьютера.

По умолчанию Windows удаляет файл подкачки после каждого сеанса работы и создает его в процессе загрузки ОС. Размер файла постоянно меняется по мере выполнения приложений и контролируется ОС. Обычно используется единственный файл подкачки, расположенный на том же диске, что и ОС. Такой подход не является лучшим, и более того, практически всегда плох. В этом случае возникает несколько проблем.

1. Память может неожиданно оказаться исчерпанной из-за того, что приложение создало на жестком диске файл большого объема или ОС без предупреждения увеличила потребности файла подкачки. Большой файл подкачки приводит к дефициту дискового пространства и к увеличению непроизводительных затрат на организацию страничного обмена.

2. Файл подкачки фрагментируется, что приводит не только к медленному считыванию жесткого диска, но и к дополнительным перемещениям считывающей головки диска, а в итоге - к существенному снижению производительности.

3. Файл подкачки фрагментируется сам по себе и очень быстро, причем так, что одна и та же область памяти может оказаться в разных местах жесткого диска. В этом случае даже отдельные приложения не могут получить доступ к памяти без нескольких обращений к диску.

4. Производительность системы падает и в том случае, если ОС установлена не на самом быстром из жестких дисков, имеющихся в компьютере.

Наличие двух жестких дисков может дать значительное преимущество при настройке файла подкачки.

Для максимально эффективного использования файла подкачки нужно так его настроить, чтобы он располагался на жестком диске в виде достаточно протяженных фрагментов (это уменьшает количество перемещений считывающей головки, радикально влияющих на производительность). Кроме того, файл подкачки необходимо периодически удалять, чтобы избежать его фрагментации.

Для установки размера файла подкачки нужно выполнить следующую последовательность действий. Щелкнуть правой клавишей мыши по значку Мой компьютер и выбрать в контекстном меню строку Свойства. На экране появится окно Свойства системы (рисунок В2.17.8). Перейти на вкладку Дополнительно (рисунок В2.17.9) и нажать кнопку Параметры в рамке Быстродействие.

Предварительно следует выбрать принцип распределения времени процессора (для оптимизации работы программ, если это пользовательский компьютер, или служб, работающих в фоновом режиме, если это сервер), следует задать режим использования памяти. Для пользовательского компьютера оптимизировать работу программ, для сервера - системного кэша.

Определение размера файла подкачки до сих пор вызывает многочисленные дискуссии.

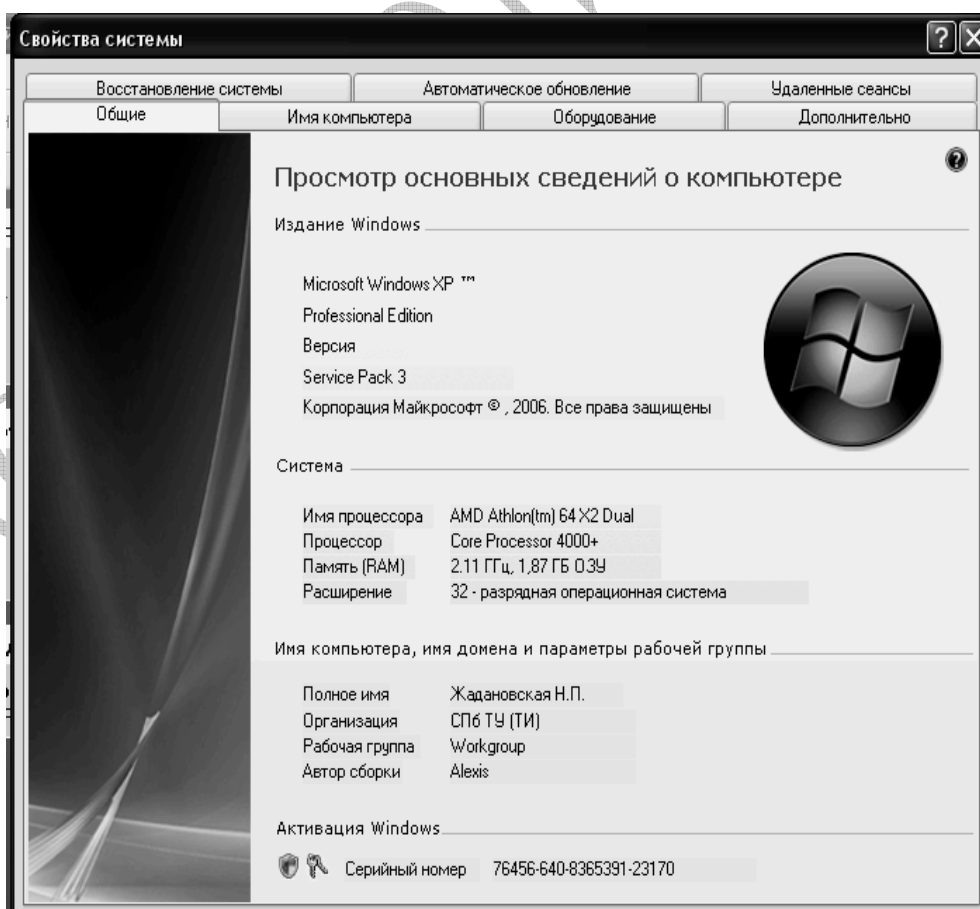
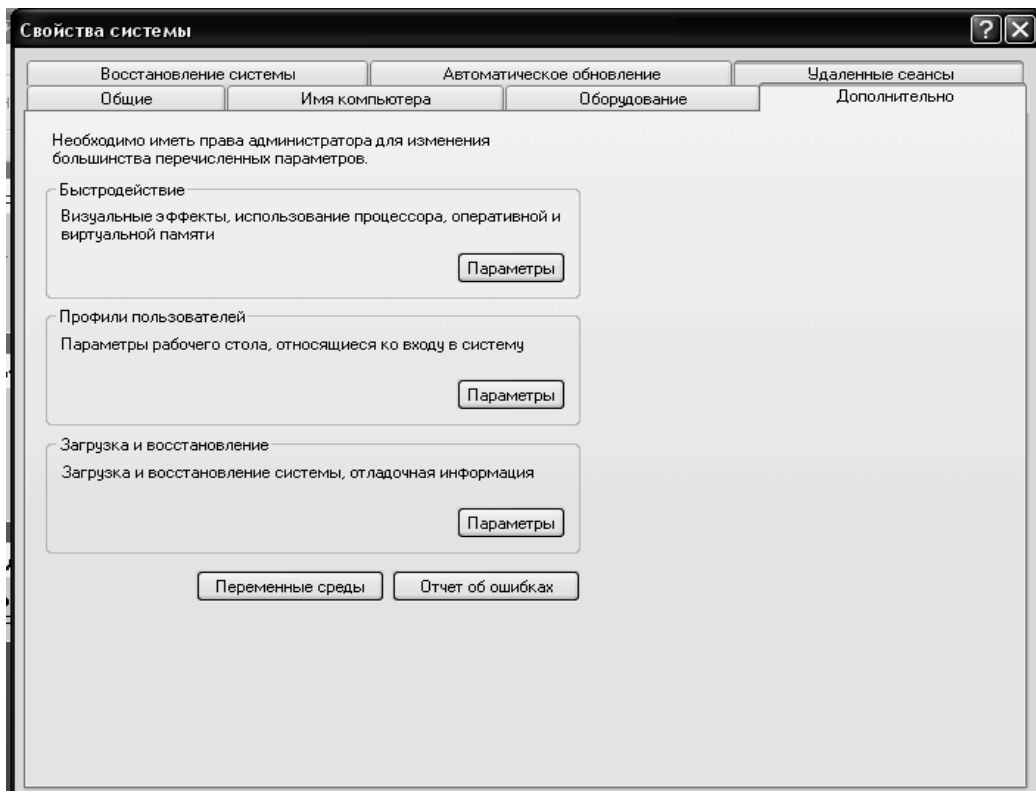
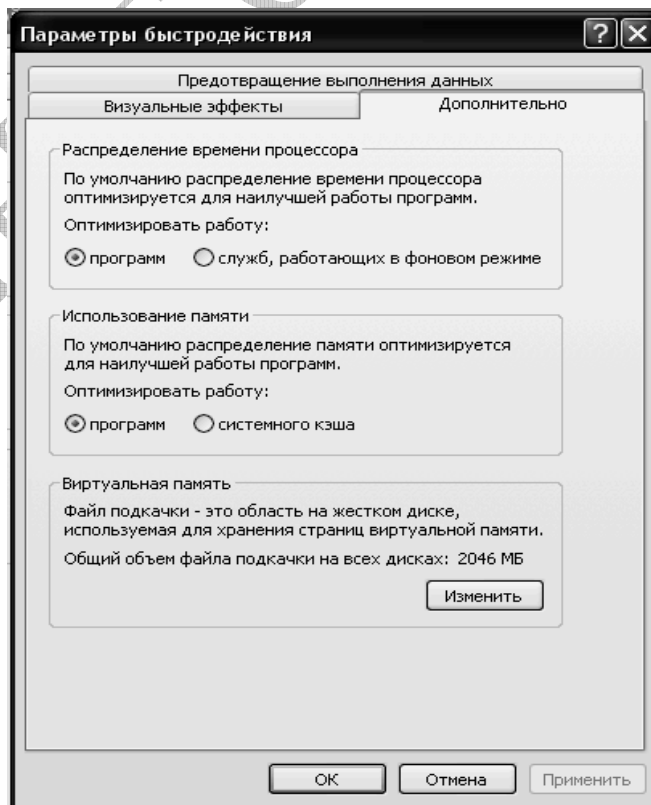


Рисунок В2.17.8 – Свойства системы



**Рисунок В217.9 – Дополнительные свойства**

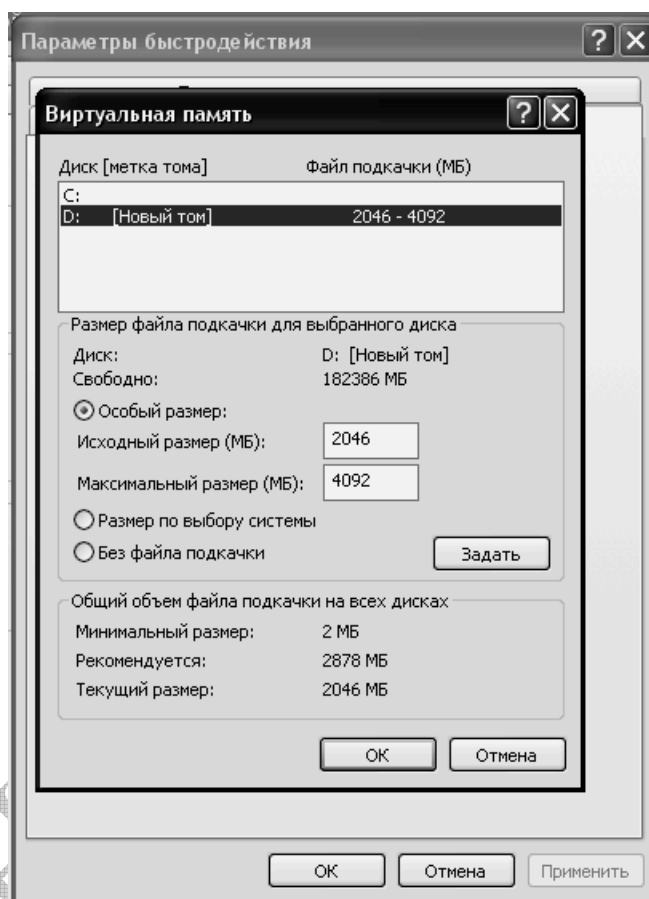
В появившемся окне Параметры быстродействия нажать кнопку Изменить (см. рисунок В2.17.10).



**Рисунок В2.17.10 – Настройки использования памяти**

Основное правило заключается в том, что при небольшом объеме оперативной памяти подкачки должен быть достаточно большим. При большом объеме оперативной памяти файл подкачки можно уменьшить.

По существующим рекомендациям, можно установить исходный размер файла подкачки, равный размеру физической памяти, а максимальный размер может быть не более двух размеров физической памяти. После этого следует нажать кнопку Задать (см. рисунок В2.17.11) и убедиться в том, что новое значение файла подкачки установлено (данное изменение требует перезагрузки компьютера).



**Рисунок В2.17.11 – Установки для файла подкачки**

Следует иметь в виду, что при первом создании файла подкачки жесткий диск, как правило, не готов к его размещению. Это обусловлено фрагментацией жесткого диска. Поэтому, нужно вначале выполнить дефрагментацию диска и лишь затем создать файл подкачки, чтобы поместить его в единственную область диска. Последовательность действий может быть, например, такой:

- если в компьютере имеется единственный жесткий диск, установить минимальный размер файла подкачки (2 Мбайт);
- если имеется два жестких диска, переместить файл подкачки на более медленный диск;
- провести дефрагментацию диска (во втором случае - быстро).
- присвоить файлу подкачки желаемый размер.

В результате работа с файлом подкачки станет максимально быстрой, а процессор и диск будут использоваться эффективно.

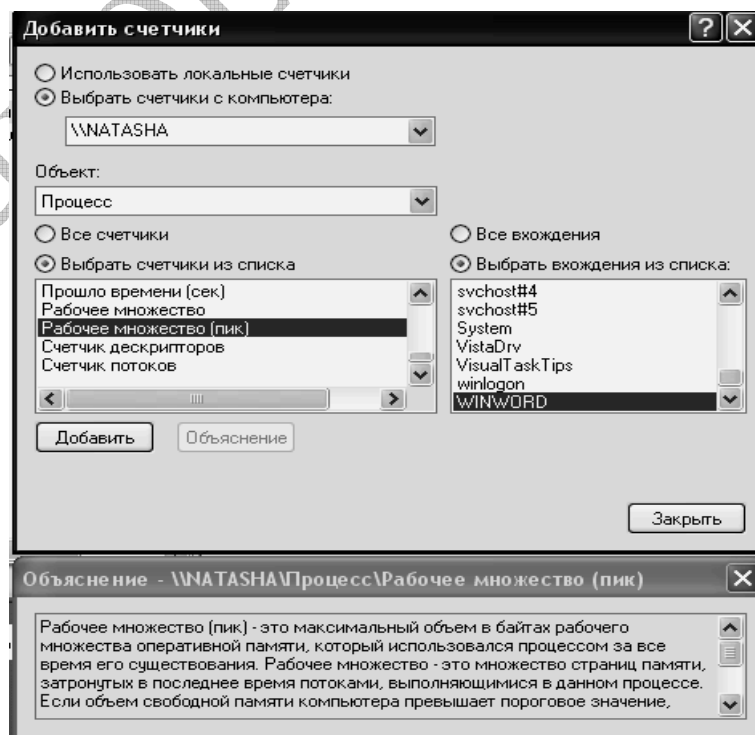
**Повышение производительности работы виртуальной памяти** сводится к определению требуемого объема физической памяти, установлению рациональной интенсивности страничного обмена и оптимизации размера и размещения файла подкачки. Основным инструментом, который может быть использован для решения этих задач, - оснастка Производительность.

Поскольку производительность компьютера главным образом определяется ресурсами процессора и памяти, следует понимать, как программы используют эти ресурсы.

Во-первых, нужно убедиться, что данная система обладает объемом памяти, рекомендуемым для работы данной ОС, т.е. необходимо сравнить имеющийся объем памяти с требованиями ОС и программ. Для этого нужно к объему памяти, необходимому ОС, добавить следующие значения:

- число пользователей, умноженное на средний размер файлов данных, открытых пользователем (для клиентского компьютера);
- число программ, запущенных на компьютере-сервере, умноженное на средний размер этих программ.

Чтобы определить, сколько памяти требуется процессу, можно включить отображение его рабочего множества в системном мониторе (см. рисунок В2.17.12), затем завершить его работу. Объем памяти, освобожденный по завершении, и будет являться объемом физической памяти, который требуется добавить в систему. Также, просмотр значений определенных счетчиков на диаграмме системного монитора может помочь в определении памяти, необходимой программам.



**Рисунок В2.17.12 – Выбор счетчика рабочее множество**

Для каждой запускаемой программы ОС выделяет часть физической памяти – рабочее множество. Эта величина может быть больше, чем минимальный объем памяти, действительно необходимый приложению. Она может отражать физический объем памяти, распределенный между несколькими процессами.

Используя счетчики, можно построить в системном мониторе диаграммы использования памяти. Значение рабочего множества представляет интерес, когда счетчик «Память\Доступно байт» (см. рисунок В2.17.13) опускается ниже определенного порога. Windows XP удовлетворяет требования программ к памяти путем использования свободных (доступных) байтов.

Когда объем свободной памяти опускается ниже определенного значения, ОС начинает пополнять его, отбирая память у рабочих множеств или менее активных программ. В результате можно заметить, что значение рабочего множества одних программ увеличивается, а других — уменьшается. Если памяти недостаточно для удовлетворения требований всех активных программ, используется файл подкачки, что снижает производительность.

Файл подкачки используется для перемещения страниц памяти, содержащих программы и данные, из оперативной памяти на диск для освобождения памяти для других целей. Использование файла подкачки - нормальное явление, т.к. это позволяет увеличить объем памяти, доступный Windows XP. Однако постоянное его использование является причиной значительного снижения производительности. Поэтому снижение использования файла подкачки может значительно ускорить работу системы.

Наблюдение за ситуациями, порождающими недостаток памяти, рекомендуется начинать со следующих счетчиков:

- Память\Доступно байт (см. рисунок В217.13);
- Память\Обмен страниц в сек (см. рисунок В217.14).

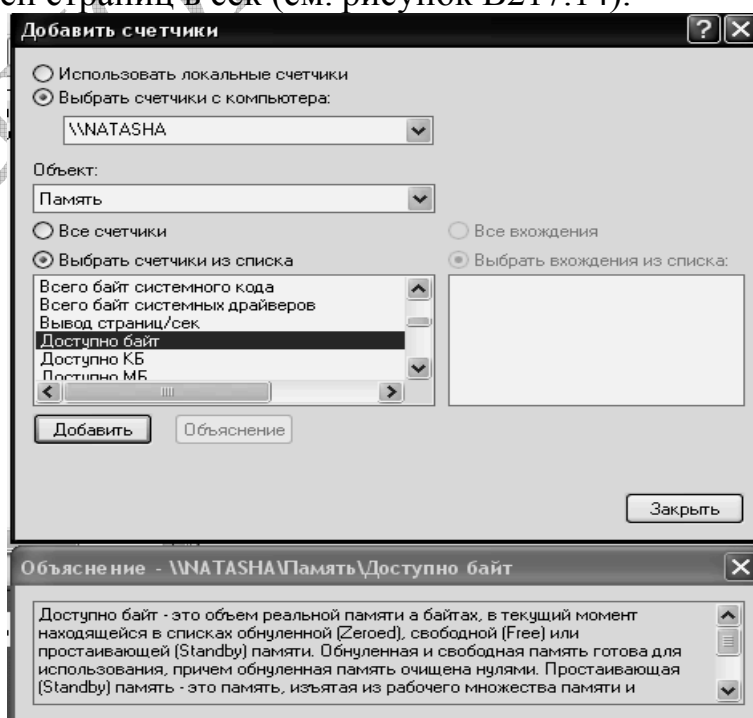


Рисунок В2.17.13 – Выбор счетчика Доступно байт





**Рисунок В2.17.14** – Выбор счетчика Обмен страниц

«Доступно байт» показывает текущий объем памяти в байтах, доступный для использования процессами. Счетчик «Обмен страниц в сек» показывает число страниц, полученных с диска из-за необходимости обращения к этим страницам или записанных на диск для освобождения свободной памяти в рабочем множестве.

Низкие значения счетчика «Доступно байт» (4 Мбайт и меньше) указывают на общий недостаток памяти на компьютере или на то, что какая-либо программа не освобождает память.

Высокие значения «Обмен страниц в сек» (превышает 20), могут не указывать на недостаток памяти, а быть результатом работы программы, использующей файл, отображенный в памяти.

Чтобы определить, является ли причиной именно это, нужно понаблюдать за счетчиками «Доступно байт», «Обмен страниц в сек» и «Файл подкачки \ % использования» (см. рисунок В2.17.15).

Детальный анализ причин возникновения недостатка памяти требует наблюдения за счетчиками «Память\Доступно байт» и «Память\Байт выделенной виртуальной памяти», чтобы отследить изменения объема памяти, и за счетчиками «Процесс\Байт исключительного использования», «Процесс\Рабочее множество» и «Процесс\Счетчик дескрипторов процессов» (см. рисунок В2.17.16), которые, как предполагается, вызывают нехватку памяти.

Так как избыточная подкачка влечет сильную загрузку жесткого диска, в результате избыточной подкачки страниц, кроме недостатка памяти, возможно также возникновение узкого места в дисковой системе.

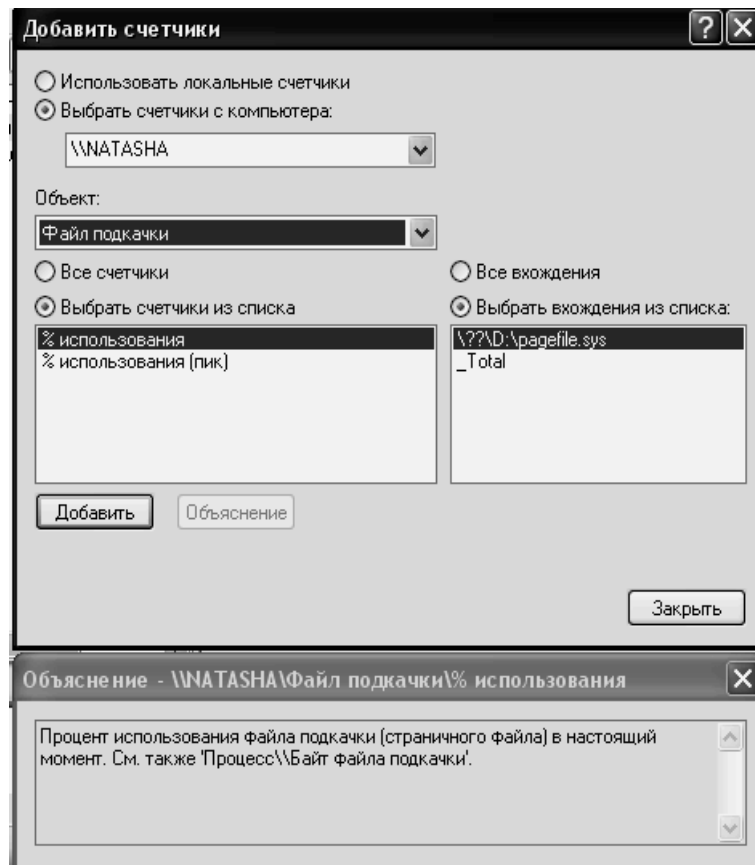


Рисунок В2.17.15 – Выбор счетчика для «Файла подкачки»

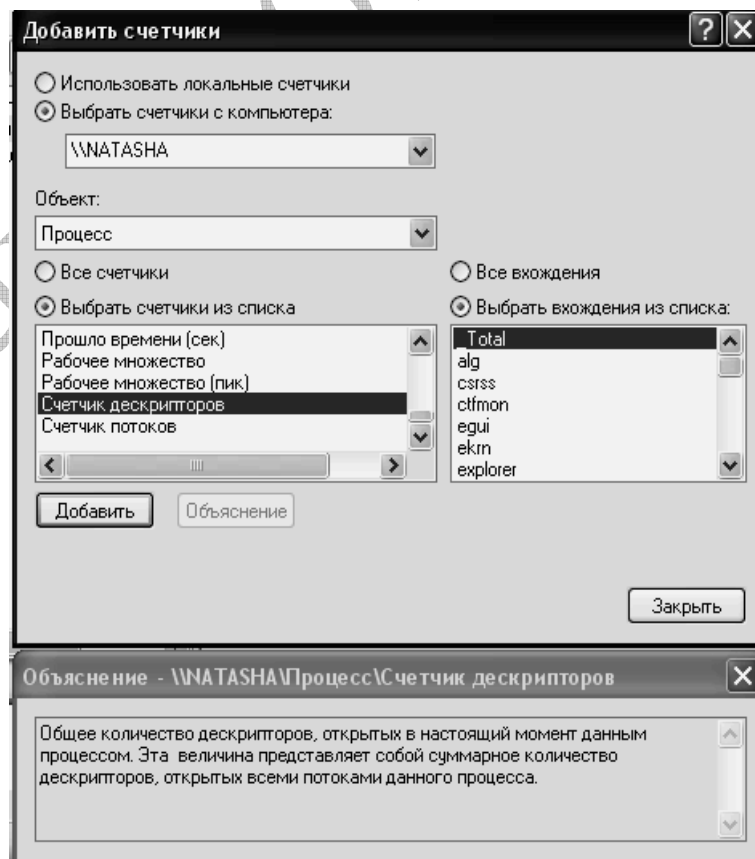


Рисунок В2.17.16 – Выбор счетчика для «Процесса»

Поэтому если при определении причины избыточной подкачки страниц недостаток памяти не прослеживается явно, следует наблюдать за счетчиками использования диска:

Логический диск\% активности диска (см. рисунок В2.17.17);

Физический \Средняя длина очереди диска.

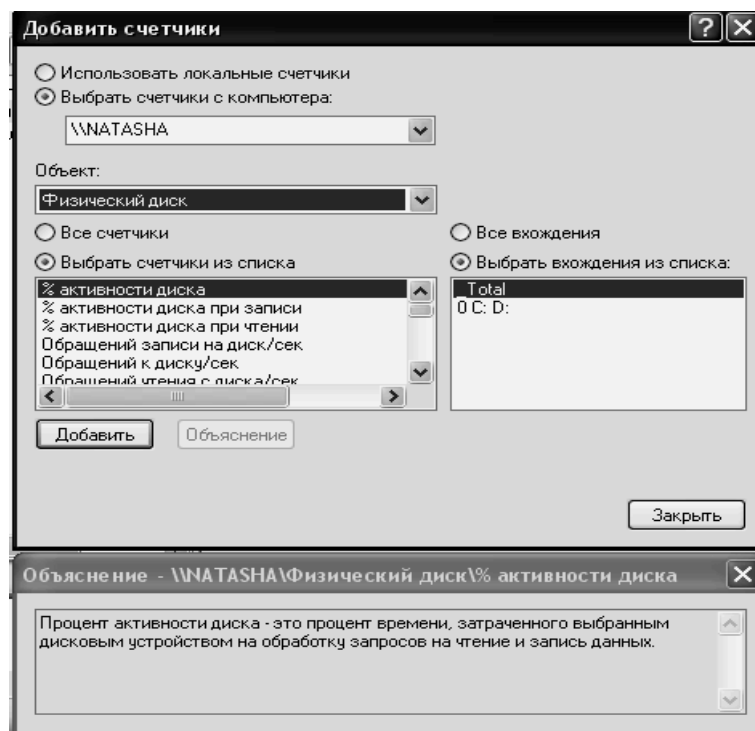


Рисунок В2.17.17 – Выбор счетчика Обмен страниц

Например, данные счетчиков «Чтение страниц/сек», «% активности диска» и «Средняя длина очереди диска», показывающие сочетание низкой активности чтения страниц с высокими значениями активности диска и средней длины очереди диска, указывают на наличие узкого места в дисковой системе. Однако если увеличение длины очереди не сопровождается уменьшением частоты чтения страниц, это означает нехватку памяти.

Чтобы определить влияние избыточной подкачки на активность диска, нужно перемножить значения счетчиков «Физический диск\Среднее время обращения к диску (сек)» и «Память\Обмен страниц в сек». Если произведение этих счетчиков превышает 0,1, подкачка занимает более 10% времени доступа к диску. Если такая ситуация наблюдается долгое время, следует нарастить объем памяти.

Целесообразно также проверить зависимость избыточной подкачки от запущенных программ. Для этого следует остановить (если возможно) работу программы, когда рабочее множество имеет наибольшее значение, и посмотреть, как при этом изменится частота подкачки страниц. При обнаружении избыточной подкачки нужно проверить значение счетчика «Память\Обмен страниц в сек». Этот счетчик показывает число страниц, которые должны быть прочитаны с диска, т.к. они отсутствуют в физической

памяти. Этот счетчик отличается от счетчика «Ошибок страницы/сек», указывающего только на то, что доступ к данным не был получен немедленно, т.к. они не были найдены в заданном рабочем множестве страниц памяти.

Существует несколько приемов, позволяющих оптимизировать использование файла подкачки для повышения производительности:

- Файл подкачки следует по возможности размещать на отдельном жестком диске.

- При наличии нескольких жестких дисков файл подкачки следует разделить, т.к. это повышает скорость работы с ним за счет того, что доступ к данным на обоих жестких дисках осуществляется одновременно.

- Однако, если имеются два жестких диска, из которых один быстрее другого, более эффективным решением будет размещение файла подкачки только на более быстром жестком диске.

Важен также размер файла подкачки. При запуске Windows XP на диске, где установлена эта ОС, автоматически создается файл подкачки (Pagefile.sys). Рекомендуется установить размер файла подкачки в 1,5 - 2 раза больше размера установленной оперативной памяти. В то же время размер этого файла также зависит от объема свободного пространства на жестком диске во время создания этого файла. Определить размер файла подкачки можно, узнав в проводнике размер файла Pagefile.sys.

Если на жестком диске еще есть свободное место, можно увеличить размер файла подкачки. Если пользователи запускают несколько программ одновременно, при увеличении размера файла подкачки их запуск может ускориться. Хотя можно оставить предложенные размеры файла подкачки, рекомендуется увеличить исходный размер, чтобы при запуске программ системе не приходилось увеличивать файла подкачки, фрагментируя его.

Когда размер файла подкачки достигает максимального, появляется сообщение о возможной остановке работы системы. Чтобы выяснить, достигает ли размер файла подкачки максимального значения, нужно сравнить реальный размер файла с его максимальным размером, который задается в окне «Свойства системы», открываемом с панели управления. Если эти значения близки, следует увеличить исходный размер файла подкачки или запускать одновременно меньшее количество приложений.

Другим способом определения оптимального значения файла подкачки является использование таких счетчиков файла подкачки, как «Файл подкачки\% использования» и «Файл подкачки\% использования (пик)». Если значение счетчика «% использования (пик)» достигает максимального размера файла подкачки или значение счетчика «% использования» близко к 100%, можно попробовать увеличить исходный размер файла подкачки.

Если файлы подкачки распределены по нескольким дискам, в качестве экземпляров счетчиков объекта «Файл подкачки» будут отображаться полные имена файлов подкачки. Можно либо добавить счетчик для каждого файла подкачки, либо выбрать экземпляр «\_Total» для наблюдения за общей активностью всех файлов подкачки.

### 3 КОНТРОЛЬНАЯ РАБОТА №3

#### **Тема: Управление процессами и ресурсами в операционной системе**

Основные определения (процесс как единица декомпозиции системы, концепция дискретных состояний процесса), подходы к программному образованию процессов, системные и пользовательские процессы, проблемы приостановки и возобновления. Взаимодействие процессов. Проблемы взаимодействия: понятие критического ресурса, синхронизация, взаимоисключение, взаимоблокировка, коммуникация между процессами, процессы в распределенных системах. Способы синхронизации процессов: семафоры, виды семафоров, мониторы, кольцевой буфер, «читатели-писатели». Уровни планирования и основные стратегии, используемые при диспетчеризации процессов: FIFO, RR, SJF, SRT. Структуры данных операционной системы для управления процессами и ресурсами. Тупики. Проблемы возникновения. Методы борьбы с тупиками. Алгоритмы обнаружения тупиков. Обход тупиков (алгоритм банкира).

#### **Содержание работы:**

##### **Задание №1 – Аналитический обзор**

Напишите ответ на поставленный вопрос. Ответ на вопросы желательно давать в развернутом виде. Вы должны показать глубину ваших знаний в ваших ответах. Эти знания помогут также выполнить остальные части контрольной работы.

##### **Задание №2 – Тестовые задания**

**2.1** Вставьте пропущенные слова

**2.2** Выберите правильные ответы (ответ м.б. не один) на предложенные тестовые задания.

##### **Задание №3 – Практические задания**

**3.1** Предложите решение теоретической задачи.

**3.2** С использованием встроенных средств современной операционной системы получите полную информацию о компонентах вычислительного процесса и проанализируйте ее.

Исследование мультипрограммного вычислительного процесса можно проводить на примере самостоятельно разработанных вами программ по курсу «Программирование на языках высокого уровня» или на приложениях, активно используемых вами на рабочем месте.

#### **Указания**

Поскольку производительность компьютера главным образом определяется ресурсами процессора и памяти, следует понимать, как программы используют эти ресурсы.

В операционной системе MS Windows XP штатными средствами необходимыми для анализа работоспособности системы являются уже знакомые по предыдущей контрольной работе диспетчер задач и системный монитор.

Теоретические знания, полученные в данном разделе курса, позволят понять многие параметры этих системных программ и оптимизировать работу вашей вычислительной системы, выявляя наиболее активные процессы, а также процессы или потоки, которые монопольно используют какие либо ресурсы или приводят к тупиковым ситуациям в системе.

Утилита Диспетчер задач в данной контрольной работе используется для просмотра текущих сведений по таким основным показателям, как список запущенных процессов и потоков и использование процессора.

Для запуска диспетчера задач можно использовать один из известных по предыдущей контрольной работе способов:

- Нажать комбинацию клавиш <Ctrl>+<Shift>+<Esc>
- Нажать комбинацию клавиш <Ctrl>+<Alt>+<Del> и при необходимости нажать в открывшемся окне кнопку Диспетчер задач (Task Manager)
- Открыть окно Выполнить (Run) и ввести команду **taskmgn**

Для просмотра запущенных процессов и показателей их производительности необходимо выбрать вкладку Процессы в окне диспетчера задач.

Инструмент, который может быть использован для определения таких явлений, как избыточный спрос на определенные ресурсы, приводящий к возникновению узкого места в работе системы - оснастка **Производительность**.

В **Примере решения** (Вариант 17) также дано более подробное руководство для выполнения этого практического задания.

Приступая к выполнению контрольных работ, рекомендуется ознакомиться со следующими методическими материалами:

1. Жадановская Н.П. Операционные системы. Базовый курс [Текст]: учебное пособие для студентов заочной формы обучения. – СПб.: СПбГТИ(ТУ), 2009. –101 с.
2. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. Учебник для ВУЗов. - СПб.: ПИТЕР, 2007.- 544с.
3. Microsoft Windows XP Home Edition и Professional/ Русские версии/ Под общей редакцией А.Н. Чекмарева. – СПб.: БХВ-Петербург, 2006. - 640с.

В качестве руководства пользователя по операционной системе и Professional м.б. использовано любое популярное руководство пользователя.

## Контрольная работа №3: Вариант 1

### 1 Аналитический обзор

Охарактеризуйте понятия систем жесткого и мягкого реального времени. Уточните, что у них общего и чем они отличаются.

### 2 Тестовые задания

**2.1** Процесс - динамический объект операционной системы, возникающий в системе после того, как пользователь или ..... решает запустить программу, то есть создать новую единицу вычислительной работы.

**2.2** В среде операционной системы с невытесняющей многозадачностью, где программа использует данные монопольно, снимаются многие проблемы:

- планирования
- управления
- разделения данных
- защиты данных
- защиты программ
- конфигурирования диска

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 5 одинаковых ресурсов R1, 5 одинаковых ресурсов R2 и 5 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.1.

Таблица В3.1 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 110                            | 555                              |
| B       | 111                            | 444                              |
| C       | 011                            | 333                              |
| D       | 101                            | 222                              |
| E       | 010                            | 111                              |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 2

### 1 Аналитический обзор

Мультипрограммные операционные системы принято разделять на системы реального времени, системы разделения времени, интерактивные, фоновые, системы пакетной обработки. Предложите для каждого из перечисленных типов работы наиболее подходящий тип планирования.

### 2 Тестовые задания

**2.1** Если ... ситуация все же возникла, часто помогает попытка начать снимать с решения некоторые задачи или заблокированные потоки.

**2.2** Дискриминацию потоков с интенсивным обменом можно компенсировать организацией дополнительной более приоритетной очереди прерванных из - за необходимости ввода- вывода ..... потоков.

- готовых
- ожидающих
- системных
- параллельных
- активных

### 3 Практические задания

**3.1** Пять задач А, В, С, D, Е поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- |                  |                  |
|------------------|------------------|
| 1) А – 12 минут, | 4) D – 2 минуты, |
| 2) В – 8 минут,  | 5) Е – 10 минут. |
| 3) С – 5 минут.  |                  |

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 10 мс,
- время кванта процессора равно – 30 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.



## Контрольная работа 3: Вариант 3

### 1 Аналитический обзор

Поясните, чем объясняется потенциально более высокая надежность операционных систем, в которых реализована вытесняющая многозадачность.

### 2 Тестовые задания

**2.1** Примеры необходимости прерываний: в оперативной памяти отсутствуют данные, необходимые активной задаче; произошло ..., когда что-то случилось, появилось или завершилось в системе; более приоритетной задаче требуется процессор.

**2.2** В мультипрограммных операционных системах на содержание дисциплины формирования очереди влияют:

- дисциплина обслуживания
- приоритеты запросов
- выбор пользователя
- дополнительные соглашения владельцев процессов
- типы и классы единиц вычислительной работы
- стратегии размещения

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 4 одинаковых ресурсов R1, 4 одинаковых ресурсов R2, 4 одинаковых ресурсов R3 и 4 одинаковых ресурсов R4. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.3.

Таблица В3.3 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов |    |    |    | Максимальная потребность |    |    |    |
|---------|------------------------|----|----|----|--------------------------|----|----|----|
|         | R1                     | R2 | R3 | R4 | R1                       | R2 | R3 | R4 |
| A       |                        | 2  | 0  | 0  | 2                        | 0  | 2  | 2  |
| B       | 2                      | 2  | 0  | 0  | 2                        | 2  | 2  | 2  |
| C       | 0                      | 2  | 2  | 0  | 2                        | 4  | 2  | 4  |
| D       | 0                      | 0  | 2  | 2  | 0                        | 0  | 2  | 4  |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 4

### 1 Аналитический обзор

Поясните употребление терминов «программа», «процесс», «задача», «поток», «нить».

### 2 Тестовые задания

**2.1** Операционная система Windows NT может повышать приоритет потока (называемый в этом случае .....), если поток не полностью использовал свой квант, или понижать его в противном случае.

**2.2** Способ реализации системных вызовов зависит от структурной организации операционной системы, связанной с особенностями:

- приоритетного обслуживания
- внешней памяти
- обработки прерываний
- аппаратной платформы
- оперативной памяти
- выбора пользователя

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 6 одинаковых ресурсов R1, 6 одинаковых ресурсов R2 и 6 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.4.

Таблица В3.4 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 210                            | 444                              |
| B       | 101                            | 222                              |
| C       | 012                            | 222                              |
| D       | 021                            | 444                              |
| E       | 201                            | 222                              |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 5

### 1 Аналитический обзор

Поясните, какими средствами синхронизации процессов располагает современная операционная система.

### 2 Тестовые задания

**2.1** Средства синхронизации могут образовывать иерархию или иметь специализацию (для синхронизации потоков одного или разных .....).

**2.2** Система действий, реализующая определенную функцию в вычислительной системе (ВС) и оформленная так, что управляющая программа ВС может перераспределять ее ресурсы в целях обеспечения мультипрограммирования, называется:

- системным вызовом
- потоком
- процессом
- программой
- процедурой

### 3 Практические задания

**3.1** Четыре задачи А, В, С, D поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- А – 8 минут,
- В – 10 минут,
- С – 2 минуты
- D – 5 минут.

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 4мс,
- время кванта процессора равно – 40 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 6

### 1 Аналитический обзор

Опишите все возможные смены состояний процесса с учетом операций приостановки и возобновления.

### 2 Тестовые задания

**2.1** Выборка потока на ..... осуществляется в соответствии с принятым в данной операционной системе правилом с учетом всех существующих в данный момент потоков и процессов.

**2.2** В основу управления процессами положена концепция дискретных состояний. Укажите состояние, которое не определено для процессов в операционной системе:

- выполнение
- синхронизация
- ожидание
- готовность
- взаимоисключение
- безопасное

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 7 одинаковых ресурсов R1, 7 одинаковых ресурсов R2, 7 одинаковых ресурсов R3 и 7 одинаковых ресурсов R4. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.6.

Таблица В3.6 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов |    |    |    | Максимальная потребность |    |    |    |
|---------|------------------------|----|----|----|--------------------------|----|----|----|
|         | R1                     | R2 | R3 | R4 | R1                       | R2 | R3 | R4 |
| A       | 3                      | 1  | 0  | 2  | 6                        | 5  | 5  | 6  |
| B       | 1                      | 0  | 1  | 1  | 3                        | 3  | 4  | 4  |
| C       | 2                      | 2  | 1  | 2  | 3                        | 4  | 3  | 3  |
| D       | 0                      | 2  | 3  | 1  | 7                        | 7  | 7  | 7  |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 7

### 1 Аналитический обзор

Поясните, являются ли синонимами понятия планирования и диспетчеризации процессов.

### 2 Тестовые задания

**2.1** Поток, сделавший синхронный (блокирующий) системный вызов, переводится планировщиком операционной системы в состояние ....., а после завершения обработки вызова - в состояние готовности.

**2.2** Операционная система также выполняет функции синхронизации процессов, приостанавливая выполнение одного из них до наступления в системе какого-либо:

- сбой
- события
- происшествия
- состояния
- времени

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 8 одинаковых ресурсов R1, 8 одинаковых ресурсов R2 и 8 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.7.

Таблица В3.7 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 321                            | 555                              |
| B       | 210                            | 222                              |
| C       | 101                            | 111                              |
| D       | 012                            | 333                              |
| E       | 123                            | 444                              |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 8

### 1 Аналитический обзор

Укажите различия между планировщиком заданий, планировщиком промежуточного уровня и диспетчером.

### 2 Тестовые задания

2.1 Разработчики операционных систем предоставляют в распоряжение ..... набор разных по своим возможностям средств синхронизации процессов и потоков.

2.2 В операционных системах реального времени после каждого прерывания планировщик- диспетчер процессов просматривает расписание и проверяет, не пора ли задачи:

- запустить
- остановить
- переключить
- сменить
- приостановить
- синхронизировать

### 3 Практические задания

3.1 Пять задач A, B, C, D, E поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- |                  |                  |
|------------------|------------------|
| 4) A – 13 минут, | 4) D – 3 минуты, |
| 5) B – 7 минут,  | 5) E – 10 минут. |
| 6) C – 6 минут.  |                  |

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 8 мс,
- время кванта процессора равно – 80 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

3.2 Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа №3: Вариант 9

### 1 Аналитический обзор

Уточните, какими средствами синхронизации процессов располагает современная операционная система.

### 2 Тестовые задания

**2.1** Поскольку посылка сигнала предусматривает знание идентификатора процесса, то взаимодействие посредством сигналов возможно только между ..... процессами, которые могут получить данные об идентификаторах друг друга.

**2.2** Ресурс, предоставляемый процессу во время его выполнения и допускающий его одновременное использование более чем одним процессом, называется ...

- параллельно используемым
- одновременным
- общим
- разделяемым
- комплексным
- взаимоисключающим

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 4 одинаковых ресурсов R1, 4 одинаковых ресурсов R2, 4 одинаковых ресурсов R3 и 4 одинаковых ресурсов R4. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.9.

Таблица В3.9 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов |    |    |    | Максимальная потребность |    |    |    |   |
|---------|------------------------|----|----|----|--------------------------|----|----|----|---|
|         | R1                     | R2 | R3 | R4 | R1                       | R2 | R3 | R4 |   |
| A       |                        |    | 1  | 2  | 3                        | 3  | 6  | 7  | 8 |
| B       | 2                      | 3  | 4  | 2  |                          |    |    |    |   |
| C       | 3                      | 4  | 1  | 2  |                          |    |    |    |   |
| D       | 4                      | 1  | 2  | 3  |                          |    |    |    |   |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 10

### 1 Аналитический обзор

Поясните, в чем состоит принципиальное отличие состояний «ожидания» (блокирования) и «готовности» процесса, ведь и в том и в другом он ожидает некоторого события.

### 2 Тестовые задания

**2.1** Число, характеризующее степень привилегированности процесса при выделении ему главного ресурса – процессора и переводе его из состояния «готовности» в состояние «выполнения» - называется ...

**2.2** Данные, отражающие состояние аппаратуры компьютера в данный момент, а также параметры операционной среды, называются:

- дескриптором
- словом состояния процессора
- контекстом
- словом состояния машины
- описанием среды

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 13 одинаковых ресурсов R1, 13 одинаковых ресурсов R2 и 13 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.10.

**Таблица В3.10** - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 420                            | 555                              |
| B       | 321                            | 555                              |
| C       | 222                            | 555                              |
| D       | 123                            | 555                              |
| E       | 024                            | 555                              |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.



## Контрольная работа 3: Вариант 11

### 1 Аналитический обзор

Проанализируйте последствие бесконечного откладывания для систем реального времени и разделения времени.

### 2 Тестовые задания

**2.1** В общем случае в операционной системе реализовано двухуровневый механизм планирования работ: на верхнем уровне работает ....., на нижнем - оставшееся процессорное время распределяется диспетчером потоков.

**2.2** При появлении в операционной системе более приоритетного готового к выполнению потока при дисциплине обслуживания с относительными приоритетами выполнение текущего потока:

- иногда не прерывается
- всегда прерывается
- не прерывается
- останавливается
- часто прерывается

### 3 Практические задания

**3.1** Четыре задачи А, В, С, D поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- А – 4 минуты,
- В – 2 минуты
- С – 7 минут
- D – 10 минут.

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 4 мс,
- время кванта процессора равно – 20 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 12

### 1 Аналитический обзор

Поясните, что такое бесконечное откладывание, чем оно отличается от тупика и что у них общего.

### 2 Тестовые задания

2.1 В схеме абсолютных приоритетов также выполняется маскирование: на время обслуживания каждого запроса вводится ..... обслуживания запросов с равным или более низким приоритетом.

2.2 В распределенных системах состоящих из нескольких процессоров (каждый из которых имеет свою оперативную память) синхронизация м.б. реализована только с помощью передачи:

- сообщения
- сигнала
- специального кода
- специального символа
- мьютекса
- семафора

### 3 Практические задания

3.1 В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 4 одинаковых ресурсов R1, 4 одинаковых ресурсов R2, 4 одинаковых ресурсов R3 и 4 одинаковых ресурсов R4. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.12.

Таблица В3.12 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов |    |    |    | Максимальная потребность |    |    |    |
|---------|------------------------|----|----|----|--------------------------|----|----|----|
|         | R1                     | R2 | R3 | R4 | R1                       | R2 | R3 | R4 |
| A       | 0                      | 1  | 1  | 1  | 3                        | 3  | 4  | 4  |
| B       | 2                      | 0  | 1  | 3  | 7                        | 7  | 7  | 7  |
| C       | 1                      | 3  | 2  | 0  | 5                        | 6  | 6  | 5  |
| D       | 2                      | 2  | 2  | 1  | 4                        | 3  | 3  | 3  |

3.2 Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 13

### 1 Аналитический обзор

Перечислите основные цели планирования процессов в мультипрограммной системе, а также поясните разницу между планировщиком и диспетчером процессов.

### 2 Тестовые задания

**2.1** Асинхронный системный вызов приводит к переходу потока в состояние ... и неясно когда поток сможет воспользоваться результатами этого системного вызова.

**2.2** Если квант станет больше, суммарные накладные (дополнительные) расходы на переключение процессов будут:

- меньше
- не меньше
- больше
- не больше
- останется прежним

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 20 одинаковых ресурсов R1, 20 одинаковых ресурсов R2 и 20 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.13.

**Таблица В3.13** - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 432                            | 888                              |
| B       | 543                            | 777                              |
| C       | 454                            | 666                              |
| D       | 345                            | 777                              |
| E       | 234                            | 888                              |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 14

### 1 Аналитический обзор

Поясните, что понимается под вытесняющими и невытесняющими алгоритмами планирования

### 2 Тестовые задания

**2.1** Задача, оформленная в виде нескольких ..... в рамках одного процесса, может быть выполнена быстрее за счет псевдопараллельного (или параллельного в мультипроцессорной системе) выполнения отдельных ее частей.

**2.2** При необходимости взаимодействия процессы обращаются к операционной системе, которая, выполняя функции посредника, предоставляет им средства межпроцессной связи. Укажите, что является такими средствами:

- конвейеры,
- кластеры
- почтовые ящики,
- разделяемые секции памяти
- логические сектора

### Практические задания

**3.1** Пять задач А, В, С, D, Е поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- |                  |                  |
|------------------|------------------|
| 7) А – 10 минут, | 4) D – 4 минуты, |
| 8) В – 8 минут,  | 5) Е – 12 минут. |
| 9) С – 2 минуты. |                  |

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 4 мс,
- время кванта процессора равно – 80 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 15

### 1 Аналитический обзор

В основу управления процессами положена концепция дискретных состояний процессов. Поясните, какие события вызывают перевод процесса из одного состояния в другое.

### 2 Тестовые задания

2.1 Так как в системах разделения времени каждой задаче выделяется только ..... процессорного времени, ни одна задача не занимает процессор надолго и время ответа оказывается приемлемым.

2.2 Укажите, каких смен состояний процессов в операционной системе не существует:

- выполнение → готовность
- ожидание → выполнение
- ожидание → готовность
- готовность → ожидание
- выполнение → ожидание

### 3 Практические задания

3.1 В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 18 одинаковых ресурсов R1, 18 одинаковых ресурсов R2 и 18 одинаковых ресурсов R3. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.15.

Таблица В3.15 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов R1R2 R3 | Максимальная потребность R1R2 R3 |
|---------|--------------------------------|----------------------------------|
| A       | 513                            | 888                              |
| B       | 424                            | 777                              |
| C       | 335                            | 666                              |
| D       | 241                            | 777                              |
| E       | 152                            | 888                              |

3.2 Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 16

### 1 Аналитический обзор

Перечислите все возможные состояния процессов в операционной системе и основные возможные операции над процессами для обеспечения смены этих состояний.

### 2 Тестовые задания

**2.1** Если ..... выбран достаточно небольшим, то у всех пользователей, одновременно работающих на одной и той же машине, складывается впечатление, что каждый из них единолично использует машину.

**2.2** Приоритет потоков, выполняемых в обычной последовательности, определяемой планировщиком потоков, (по сравнению с приоритетом обработчиков прерываний) в общем случае:

- всегда ниже
- не выше
- всегда выше
- не ниже
- равен ему

### 3 Практические задания

**3.1** В контексте «алгоритма банкира» определите и обоснуйте, является ли приведенное состояние опасным или безопасным с точки зрения возникновения тупиков.

Предположим, что в системе имеются 4 одинаковых ресурсов R1, 4 одинаковых ресурсов R2, 4 одинаковых ресурсов R3 и 4 одинаковых ресурсов R4. Текущее распределение ресурсов и максимальное их количество, необходимое процессам представлено в таблице В3.16.

Таблица В3.16 - Текущее распределение ресурсов

| Процесс | Предоставлено ресурсов |    |    |    | Максимальная потребность |    |    |    |
|---------|------------------------|----|----|----|--------------------------|----|----|----|
|         | R1                     | R2 | R3 | R4 | R1                       | R2 | R3 | R4 |
| A       | 3                      | 2  | 2  | 4  | 4                        | 3  | 3  | 4  |
| B       | 2                      | 1  | 3  | 5  | 6                        | 5  | 8  | 7  |
| C       | 1                      | 4  | 3  | 2  | 5                        | 8  | 7  | 6  |
| D       | 4                      | 3  | 2  | 1  | 7                        | 8  | 5  | 6  |

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 17 – Примеры решения

### 1 Аналитический обзор

Очень важным понятием синхронизации процессов является понятие «критической секции», которая определяется по отношению к определенным критическим данным. Поясните, что под этим понимается.

### 2 Тестовые задания

**2.1** Другой важной концепцией, лежащей в основе многих вытесняющих алгоритмов планирования, является ..... обслуживание, которое предполагает наличие у потоков некоторой изначально известной характеристики.

**2.2** Командный язык операционной системы OS/2 позволяет задать динамически изменяемую величину кванта времени (команда TIMESLICE):

- в любом диапазоне
- в диапазоне, ограниченном пользователем
- в случайном диапазоне
- в диапазоне, ограниченном системой
- в зависимости от числа запущенных в системе процессов и потоков

### 3 Практические задания

**3.1** Три задачи А, В, С поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- А – 4 минуты,
- В – 2 минуты
- С - 7 минут.

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 2 мс,
- время кванта процессора равно – 20 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

**3.2** Запустите выбранное для целей исследования приложение.

С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.

Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.

## Контрольная работа 3: Вариант 17 – Примеры решения

### 1 Аналитический обзор

**Напишите ответ на поставленный вопрос:**

**Очень важным понятием синхронизации процессов является понятие «критической секции», которая определяется по отношению к определенным критическим данным. Поясните, что под этим понимается.**

Важным понятием синхронизации как процессов, так и потоков является понятие «критической секции» программы. Критическая секция — это часть программы, результат выполнения которой может непредсказуемо меняться, если переменные, относящиеся к этой части программы, изменяются другими потоками в то время, когда выполнение этой части еще не завершено.

Критическая секция всегда определяется по отношению к определенным критическим данным, при несогласованном изменении которых могут возникнуть нежелательные эффекты (примерами критических данных могут являться записи файла корпоративной базы данных).

Во всех потоках, работающих с критическими данными, должна быть определена критическая секция.

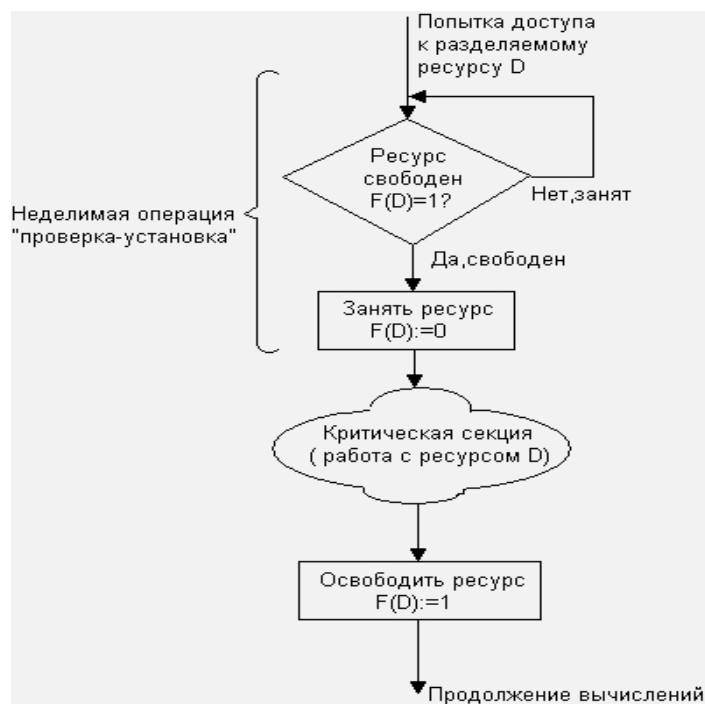
По отношению к критическим данным, необходимо обеспечить, чтобы в каждый момент времени в критической секции, связанной с этими данными, находился только один поток. При этом неважно, находится этот поток в активном или в приостановленном состоянии. Этот прием называют взаимным исключением.

**Операционная система – ОС** - использует разные способы реализации взаимного исключения. Некоторые способы пригодны для взаимного исключения при вхождении в критическую секцию только потоков одного процесса, в то время как другие могут обеспечить взаимное исключение и для потоков разных процессов.

Для синхронизации потоков одного процесса прикладной программист может использовать глобальные блокирующие переменные. С этими переменными, к которым все потоки процесса имеют прямой доступ, программист работает, не обращаясь к системным вызовам ОС. Каждому набору критических данных ставится в соответствие двоичная переменная, которой поток присваивает значение 0, когда он входит в критическую секцию, и значение 1, когда он ее покидает.

На рисунке В17.1 показан фрагмент алгоритма потока, использующего для реализации взаимного исключения доступа к критическим данным  $D$  блокирующую переменную  $F(D)$ . Перед входом в критическую секцию поток проверяет, не работает ли уже какой-нибудь поток с данными  $D$ . Если переменная  $F(D)$  установлена в 0, то данные заняты и проверка циклически повторяется. Если же данные свободны ( $F(D) = 1$ ), то значение переменной  $F(D)$  устанавливается в 0 и поток входит в критическую секцию. После того как поток выполнит все действия с данными  $D$ , значение переменной  $F(D)$  снова устанавливается равным 1.





**Рисунок В3.17.1** - Реализация критических секций с использованием блокирующих переменных

Блокирующие переменные могут использоваться не только при доступе к разделяемым данным, но и при доступе к разделяемым ресурсам любого вида.

Если все потоки написаны с учетом вышеописанных соглашений, то взаимное исключение гарантируется. При этом потоки могут быть прерваны ОС в любой момент и в любом месте, в том числе в критической секции.

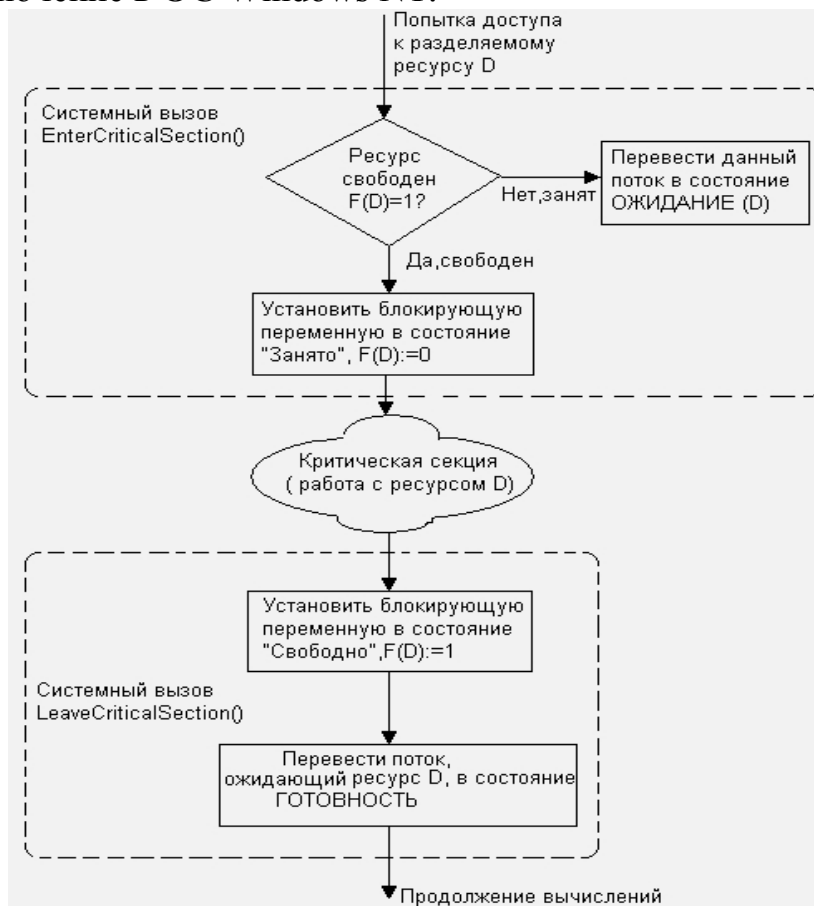
Однако следует заметить, что одно ограничение на прерывания все же имеется. Нельзя прерывать поток между выполнением операций проверки и установки блокирующей переменной. Поясним это. Пусть в результате проверки переменной поток определил, что ресурс свободен, но сразу после этого, не успев установить переменную в 0, был прерван. За время его приостановки другой поток занял ресурс, вошел в свою критическую секцию, но также был прерван, не завершив работы с разделяемым ресурсом. Когда управление было возвращено первому потоку, он, считая ресурс свободным, установил признак занятости и начал выполнять свою критическую секцию. Таким образом, был нарушен принцип взаимного исключения, что потенциально может привести к нежелательным последствиям. Во избежание таких ситуаций в системе команд многих компьютеров предусмотрена единая, неделимая команда анализа и присвоения значения логической переменной (например, команды BTC, BTR и BT5 процессора Pentium). При отсутствии такой команды в процессоре соответствующие действия должны реализовываться специальными системными примитивами, которые бы запрещали прерывания на протяжении всей операции проверки и установки.

Реализация взаимного исключения описанным выше способом имеет существенный недостаток: в течение времени, когда один поток находится в критической секции, другой поток, которому требуется тот же ресурс, получив

доступ к процессору, будет непрерывно опрашивать блокирующую переменную, бесполезно тратя выделяемое ему процессорное время, которое могло бы быть использовано для выполнения другого потока.

Для устранения этого недостатка во многих ОС предусматриваются специальные системные вызовы для работы с критическими секциями.

На рисунке В17.2 показано, как с помощью этих функций реализовано взаимное исключение в ОС Windows NT.



**Рисунок В3.17.2** - Реализация взаимного исключения с использованием системных функций входа в критическую секцию и выхода из нее

Перед тем как начать изменение критических данных, поток выполняет системный вызов EnterCriticalSection(). В рамках этого вызова сначала выполняется, как и в предыдущем случае, проверка блокирующей переменной, отражающей состояние критического ресурса. Если системный вызов определил, что ресурс занят ( $F(D) = 0$ ), он в отличие от предыдущего случая не выполняет циклический опрос, а переводит поток в состояние ожидания и делает отметку о том, что данный поток должен быть активизирован, когда соответствующий ресурс освободится.

Поток, который в это время использует данный ресурс, после выхода из критической секции должен выполнить системную функцию LeaveCriticalSection(), в результате чего блокирующая переменная принимает значение, соответствующее свободному состоянию ресурса ( $F(D) = 1$ ), а ОС просматривает очередь ожидающих этот ресурс потоков и переводит первый поток из очереди в состояние готовности.

Т.о. исключается непроизводительная потеря процессорного времени на циклическую проверку освобождения занятого ресурса. Однако в тех случаях, когда объем работы в критической секции небольшой и существует высокая вероятность в очень скором доступе к разделяемому ресурсу, более предпочтительным может оказаться использование блокирующих переменных. Действительно, в такой ситуации накладные расходы ОС по реализации функции входа в критическую секцию и выхода из нее могут превысить полученную экономию.

Обобщением блокирующих переменных являются так называемые семафоры Дейкстры. Вместо двоичных переменных Дейкстра (Dijkstra) предложил использовать переменные, которые могут принимать целые неотрицательные значения. Такие переменные, используемые для синхронизации вычислительных процессов, получили название семафоров.

Для работы с семафорами вводятся два примитива, традиционно обозначаемых P и V. Пусть переменная S представляет собой семафор. Тогда действия V(S) и P(S) определяются следующим образом.

- **V(S):** переменная S увеличивается на 1 единым действием. Выборка, наращивание и запоминание не могут быть прерваны. К переменной S нет доступа другим потокам во время выполнения этой операции.
- **P(S):** уменьшение S на 1, если это возможно. Если  $S=0$  и невозможно уменьшить S, оставаясь в области целых неотрицательных значений, то в этом случае поток, вызывающий операцию P, ждет, пока это уменьшение станет возможным. Успешная проверка и уменьшение также являются неделимой операцией.

Никакие прерывания во время выполнения примитивов V и P недопустимы. В частном случае, когда семафор S может принимать только значения 0 и 1, он превращается в блокирующую переменную, которую по этой причине часто называют двоичным семафором. Операция P включает в себе потенциальную возможность перехода потока, который ее выполняет, в состояние ожидания, в то время как операция V может при некоторых обстоятельствах активизировать другой поток, приостановленный операцией P.

Использование двоичной переменной не позволяет организовать доступ к критическому ресурсу более чем одному потоку. Семафор же решает задачу синхронизации более гибко, допуская к разделяемому пулу ресурсов заданное количество потоков. Таким образом, семафоры позволяют эффективно решать задачу синхронизации доступа к ресурсным пулам, таким, например, как набор идентичных в функциональном назначении внешних устройств (модемов, принтеров, портов), или набор областей памяти одинаковой величины, или информационных структур. Во всех этих и подобных им случаях с помощью семафоров можно организовать доступ к разделяемым ресурсам сразу нескольких потоков. Семафор может использоваться и в качестве блокирующей переменной.

В рассмотренном выше примере, для того чтобы исключить коллизии при работе с разделяемой областью памяти, будем считать, что запись в буфер и считывание из буфера являются критическими секциями. Взаимное

исключение будем обеспечивать с помощью двоичного семафора  $b$  (рисунок В3.17.3). Оба потока после проверки доступности буферов должны выполнить проверку доступности критической секции.

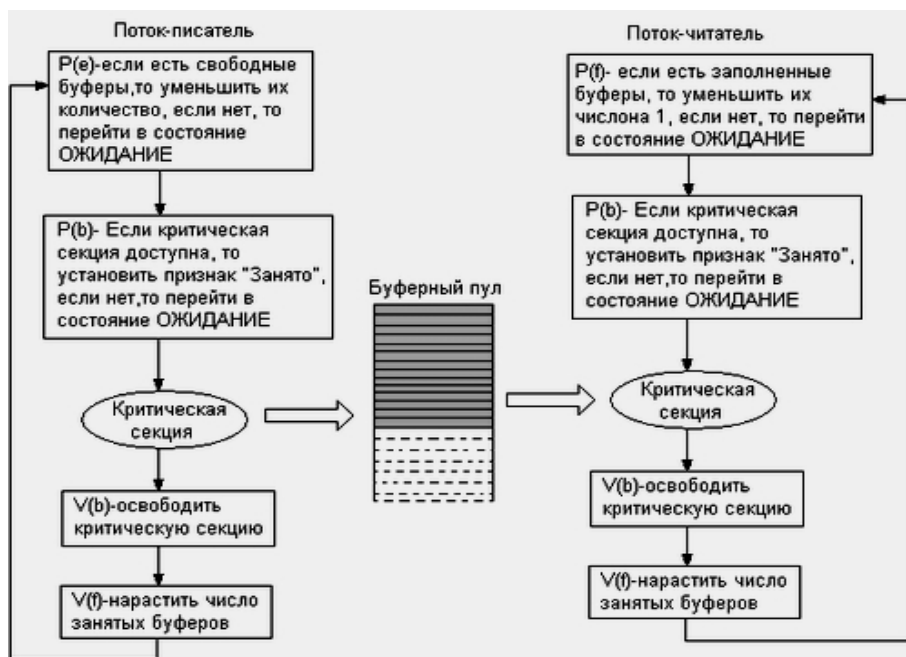


Рисунок В17.3 - Использование двоичного семафора

## 2 Тестовые задания

### 2.1 Вставьте пропущенное слово:

Другой важной концепцией, лежащей в основе многих вытесняющих алгоритмов планирования, является ..... обслуживание, которое предполагает наличие у потоков некоторой изначально известной характеристики.

Правильным ответом на данный вопрос является **приоритетное обслуживание**. Приоритетное обслуживание предполагает наличие у процессов такой характеристики, как приоритет, на основании которой определяется порядок их выполнения. Приоритет — это число, характеризующее степень привилегированности процесса при использовании ресурсов вычислительной машины, в частности процессорного времени: чем выше приоритет, тем выше привилегии, тем меньше времени будет проводить процесс в очередях. Приоритет может выражаться целым или дробным, положительным или отрицательным значением. В некоторых ОС принято, что приоритет потока тем выше, чем больше число, обозначающее приоритет. В других системах, наоборот, чем меньше число, тем выше приоритет. Приоритет процесса назначается ОС при его создании, при этом ОС учитывает, является этот процесс системным или прикладным, каков статус пользователя, запустившего процесс, и т.д. Во многих ОС предусматривается возможность изменения приоритетов в течение жизни процесса. В этом случае приоритеты называются динамическими в отличие от неизменяемых, фиксированных, приоритетов.

От того, какие приоритеты назначены процессам, существенно зависит эффективность работы всей вычислительной системы. В современных ОС во избежание разбалансировки системы, которая может возникнуть при неправильном назначении приоритетов, возможности пользователей влиять на приоритеты процессов и потоков стараются ограничивать.

**2.2 Выберите все правильные ответы на предложенное тестовое задание:**

**Командный язык операционной системы OS/2 позволяет задать динамически изменяемую величину кванта (команда TIMESLICE):**

- в любом диапазоне
- в диапазоне, ограниченном пользователем
- в случайном диапазоне
- в диапазоне, ограниченном системой
- в зависимости от числа потоков
- в зависимости от числа запущенных в системе процессов и потоков

Правильным ответом является - **в диапазоне, ограниченном системой**. Во многих операционных системах – ОС - алгоритмы планирования построены с использованием как концепции квантования, так и приоритетов, т.е. в основе планирования лежит квантование, но порядок выбора из очереди готовых определяется приоритетами потоков. Именно так реализовано планирование в ОС Windows NT, в которой квантование сочетается с динамическими абсолютными приоритетами. В ОС UNIX System V Release 4 понятие «поток» отсутствует, и планирование осуществляется на уровне процессов. В этой системе реализована вытесняющая многозадачность, основанная на использовании приоритетов и квантования. Каждый процесс в зависимости от задачи, которую он решает, относится к одному из трех определенных в системе приоритетных классов: классу реального времени, классу системных процессов или классу процессов разделения времени. Назначение и обработка приоритетов выполняются для разных классов по-разному. Для каждого уровня приоритета по умолчанию имеется своя величина кванта времени. Планирование в ОС OS/2 основано на использовании квантования и абсолютных динамических приоритетов. На множестве потоков определены приоритетные классы — критический (time critical), серверный (server), стандартный (regular) и остаточный (idle), в каждом из которых имеется 32 приоритетных уровня. Потоки критического класса имеют наивысший приоритет. Поток из менее приоритетного класса не может быть выбран для выполнения, пока в очереди более приоритетного класса имеется хотя бы один поток. Внутри каждого класса потоки выбираются также по приоритетам. ОС динамически устанавливает величину кванта, отводимого потоку для выполнения.

Величина кванта зависит от загрузки системы а параметры настройки системы позволяют явно задать границы изменения кванта (он не может быть меньше 32 мс и больше 65 536 мс).

### 3 Практические задания

#### 3.1 Предложите решение теоретической задачи.

Три задачи А, В, С поступают в компьютерный центр практически одновременно.

Ожидается, что время выполнения этих задач составит для:

- А – 4 минуты,
- В – 2 минуты
- С - 7 минут.

Требуется определить среднее время выполнения запущенных задач, считая, что:

- время смены контекста или время переключения между процессами равно - 2 мс,
- время кванта процессора равно – 20 мс.
- используемая дисциплина планирования – RR, при которой по истечении определенного кванта времени процесс прерывается и помещается в конец очереди готовых процессов, а процессор выделяется для использования процессу, находящемуся в ее начале.

Модификацией алгоритма FCFS является алгоритм, получивший название Round Robin (Round Robin – это вид детской карусели в США) или сокращенно RR. По сути дела, это тот же самый алгоритм, только реализованный в режиме вытесняющего планирования.

Можно представить себе все множество готовых процессов организованным циклически – процессы сидят на карусели. Карусель вращается так, что каждый процесс находится около процессора небольшой фиксированный квант времени, обычно 10 – 100 миллисекунд (см. рисунок В3.17.4).

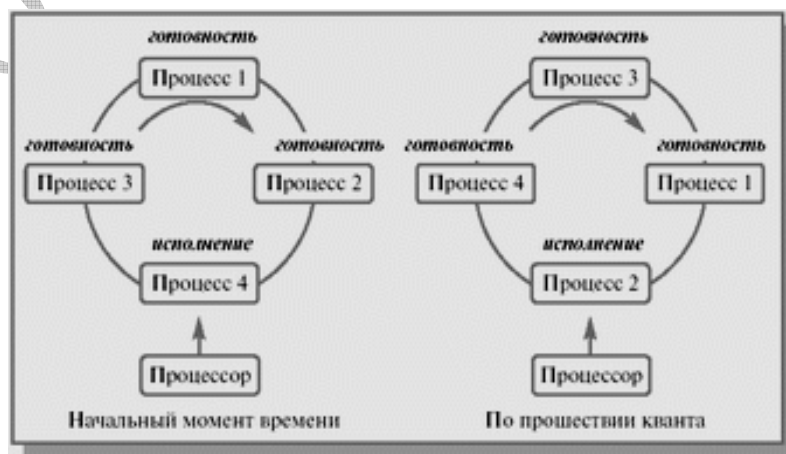


Рисунок В3.17.4 - Процессы на карусели

Пока процесс находится рядом с процессором, он получает процессор в свое распоряжение и может исполняться.

Реализуется такой алгоритм с помощью организации процессов, находящихся в состоянии готовности, в очередь FIFO.

Планировщик выбирает для очередного исполнения процесс, расположенный в начале очереди, и устанавливает таймер для генерации прерывания по истечении определенного кванта времени.

При выполнении процесса возможны два варианта.

- Время непрерывного использования процессора, необходимое процессу (остаток текущего CPU burst), меньше или равно продолжительности кванта времени. Тогда процесс по своей воле освобождает процессор до истечения кванта времени, на исполнение поступает новый процесс из начала очереди, и таймер начинает отсчет кванта заново.
- Продолжительность остатка текущего CPU burst процесса больше, чем квант времени. Тогда по истечении этого кванта процесс прерывается таймером и помещается в конец очереди процессов, готовых к исполнению, а процессор выделяется для использования процессу, находящемуся в ее начале.

Рассмотрим пример с порядком процессов А, В, С и величиной кванта времени равной 1 мин. Выполнение этих процессов иллюстрируется таблицей В3.17.1. Обозначение "И" используется в ней для процесса, находящегося в состоянии исполнения, обозначение "Г" – для процессов в состоянии готовности, пустые ячейки соответствуют завершившимся процессам.

Состояния процессов показаны на протяжении соответствующей единицы времени, т. е. колонка с номером 1 соответствует промежутку времени от 0 до 1.

**Таблица В3.17.1** Выполнение процессов

| Время | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|-------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|
| А     | И | Г | Г | И | Г | Г | И | Г | И |    |    |    |    |    |
| В     |   | Г | И | Г | Г | И |   |   |   |    |    |    |    |    |
| С     |   | Г | Г | И | Г | Г | И | Г | И | Г  | И  | И  | И  |    |

Первым для исполнения выбирается процесс А. Продолжительность его CPU burst больше, чем величина кванта времени, и поэтому процесс исполняется до истечения кванта, т. е. в течение 1 единицы времени.

Следующим начинает выполняться процесс В.. После этого он помещается в конец очереди готовых к исполнению процессов, которая принимает вид С, А, В.

Следующим начинает выполняться процесс С.. После этого он помещается в конец очереди готовых к исполнению процессов, которая принимает вид А, В, С.

Далее действия повторяются.

На втором цикле процесс В должен закончиться и дальше переключения будут уже между двумя процессами.

Время ожидания для процесса А (количество символов "Г" в соответствующей строке) составляет 5 единиц времени, для процесса В – 3 единицы времени, для процесса С – 6 единиц времени.

Таким образом, среднее время ожидания для этого алгоритма получается равным

$$(5 + 3 + 6)/3 = 4.7 \text{ (единицы времени)}$$

Полное время выполнения для процесса А (количество непустых столбцов в соответствующей строке) составляет 9 единиц времени, для процесса В – 5 единиц, для процесса С – 13 единиц.

Среднее полное время выполнения оказывается равным

$$(9 + 5 + 13)/3 = 9 \text{ (единиц времени)}$$

Как можно заметить, на производительность алгоритма RR сильно влияет величина кванта времени.

При очень малых величинах создается иллюзия того, что каждый из  $n$  процессов работает на собственном виртуальном процессоре с производительностью  $\sim 1/n$  от производительности реального процессора.

Правда, это справедливо лишь при теоретическом анализе при условии пренебрежения временами переключения контекста процессов.

В реальных условиях при слишком малой величине кванта времени и, соответственно, слишком частом переключении контекста накладные расходы на переключение резко снижают производительность системы.

Исходя из условий задачи и учитывая время переключения между процессами можно определить продолжительность цикла на первом этапе выполнения задач А, В, С

$$T_{1ц} = 20*3 + 2*3 = 66 \text{ (мс)}$$

Первой будет решена задача В. Время ее решения  $T_{в} = T_{1б}$ , с учетом времени решения задачи  $T_{зб}$  в монопольном режиме и кванта времени  $T_{кв}$ , можно найти следующим образом :

$$T_{1б} = (T_{зб}/T_{кв}) * T_{ц} = (2*20*1000 / 20) * 66 = 39600 \text{ (мс)} = 396 \text{ (с)} = 6.6 \text{ (мин)}$$

Итак, первый этап завершается решением задачи В, его продолжительность равна 6,6 мин. Задачи А и С на этом этапе также решались в течение двух минут.

На втором этапе решаются уже две задачи, поэтому  $T_{2ц}$

$$T_{2ц} = 20*2 + 2*2 = 44 \text{ (мс)}$$



Далее из задач А и С будет решена задача А. Время ее решения на втором этапе  $T_{2a}$  будет следующим:

$$T_{2a} = (T_{3a} - T_{3b}) / T_{kv} * T_{2c} = ((4-2) * 60 * 1000 / 20) * 44 = 264000 \text{ (мс)} = 264 \text{ (сек)} = 4,4 \text{ (мин)}$$

Полное время решения задачи А с учетом первого и второго этапа  $T_a$ :

$$T_a = T_{2a} + T_b = 6,4 + 4,4 = 10,8$$

После завершения второго этапа решается в режиме квантования только задача С. Продолжительность цикла на третьем этапе выполнения задач  $T_{3c}$

$$T_{3c} = 20 + 2 = 22 \text{ (мс)}$$

Задача С завершится через:

$$T_{3c} = ((T_{3c} - T_{3a}) / T_{kv} * T_{3c} = ((7-4) * 60 * 1000 / 20) * 22 = 198000 \text{ (мс)} = 198 \text{ (с)} = 3,3 \text{ (мин)}$$

А полное время решения задачи составит значение  $T_c$ :

$$T_c = T_a + T_{3c}$$

Итак, среднее время  $T_{cp}$  решения задач А, В, С

$$T_{cp} = (T_a + T_b + T_c) / 3 = (11 + 6,6 + 14,3) / 3 = 10,63 \text{ мин}$$

### 3.2 Выполните предложенное практическое задание

**Запустите выбранное для целей исследования приложение. С использованием штатных средств операционной системы MS Windows XP исследуйте образованный процесс: определите PID, базовый приоритет, количество потоков. Уточнить остальные имеющиеся характеристики процесса.**

**Измените приоритет процесса и установите, влияет ли это на время выполнения приложения.**

В качестве тестовой системы для выполнения практического задания использовалась: **операционная система – ОС- MS Windows XP PRO SP3.**

В качестве исследуемого приложения был выбран редактор MSWord. Выполнимая операция – изменение шрифта в книге. Базовый формат Times New Roman, размер шрифта – 12, кол. Страниц – 500.

В диспетчере задач MS Windows виден созданный процесс (рисунок В3.17. 5).

Диспетчер задач позволяет получить обобщенную информацию об использовании основных ресурсов компьютера: общее количество процессов и потоков, участвующих в системе и т.д. (см. рисунок В3.17.6)

В диспетчере задач отображаются сведения о программах и процессах, выполняемых на компьютере. Кроме того, там можно просмотреть наиболее часто используемые показатели быстродействия процессов.



Рисунок В3.17.5 – Стандартное окно диспетчера задач

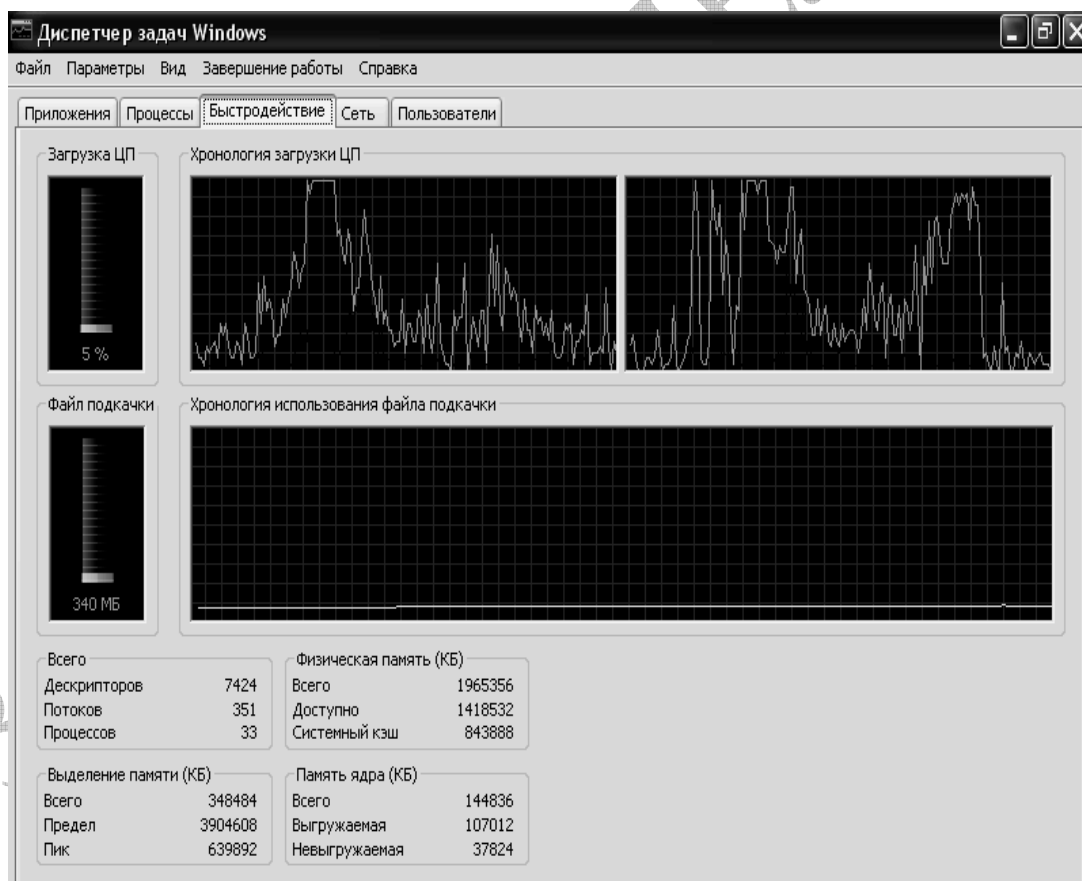


Рисунок В3.17.6 – Использование основных ресурсов компьютера

Просмотр (мониторинг) процессов осуществляется переходом на вкладку процессы. Выбрать просматриваемые характеристики можно с помощью команды Выбрать столбцы (см. рисунок В3.17.7)

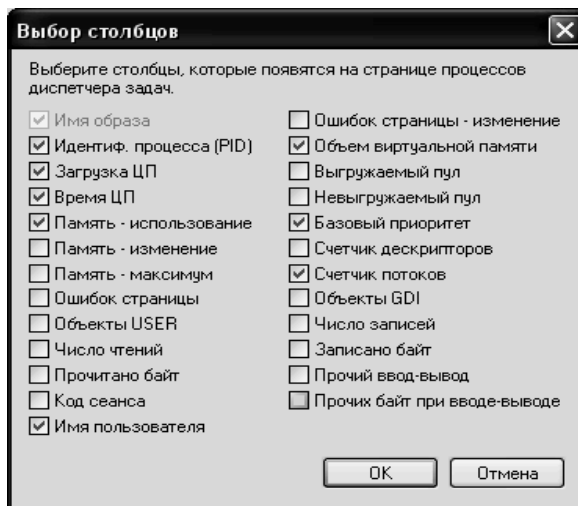


Рисунок В3.17.7 – Характеристики процессов

На рисунке В3.17.8 показан запущенный процесс и выбранные для него характеристики.

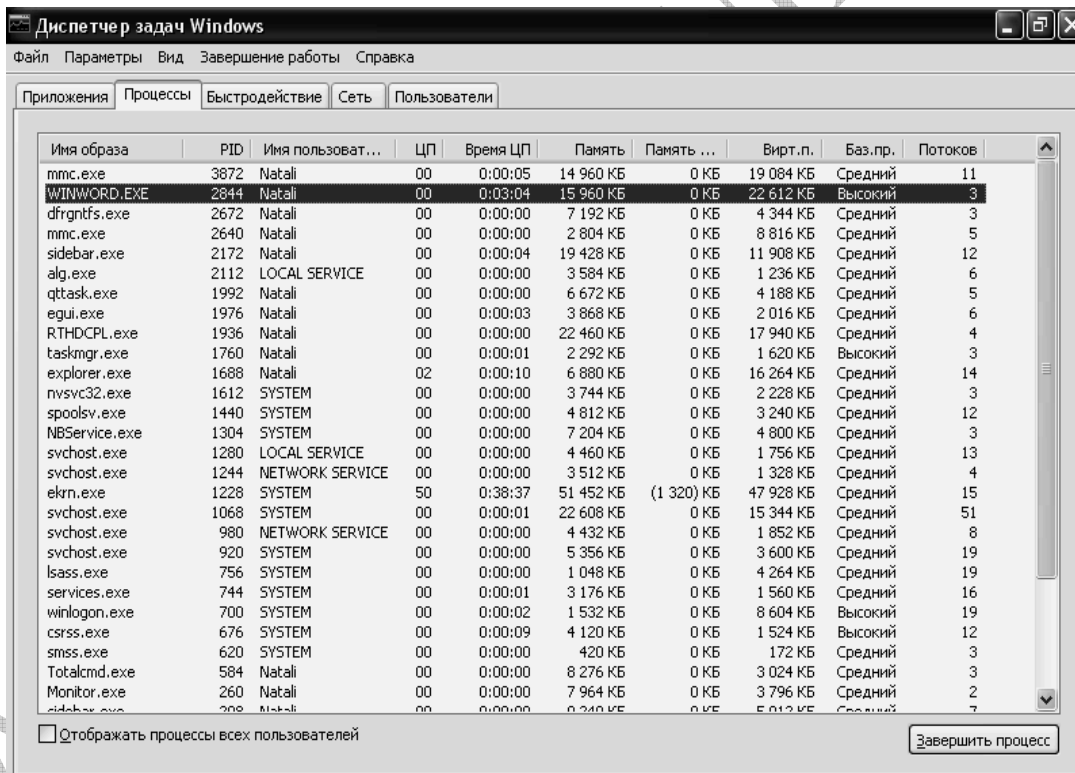
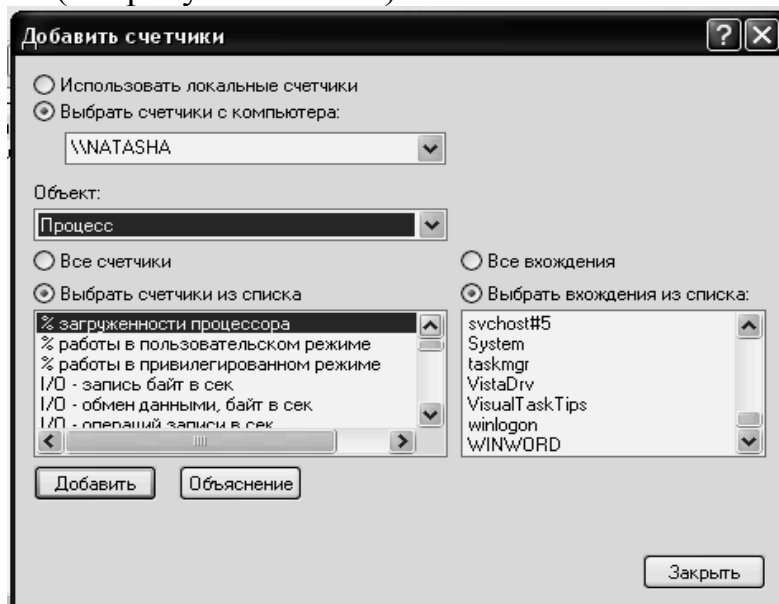


Рисунок В3.17.8 – Характеристики запущенного процесса

Системный монитор служит для сбора и просмотра в реальном времени данных памяти, диска, процессора, сети и других параметров в виде графика, гистограммы или отчета.

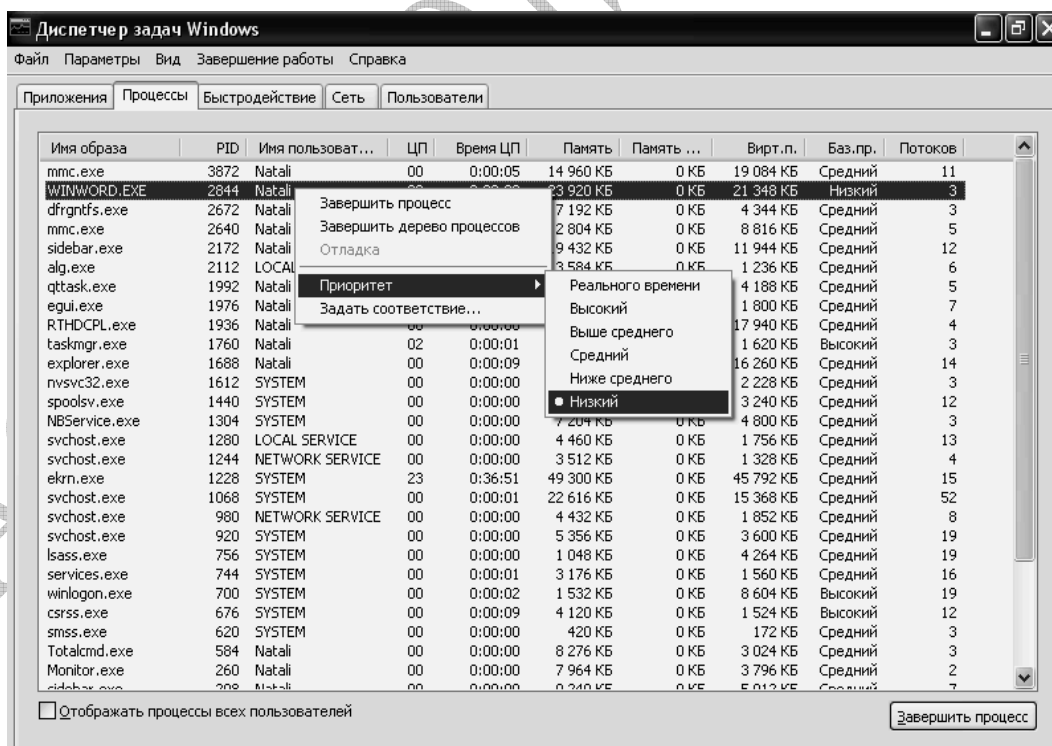
Анализ данных наблюдения позволяет обнаружить такие явления, как избыточный спрос на определенные ресурсы, приводящий к возникновению узкого места в работе системы.

Здесь также есть возможность выбрать параметры, характеризующие запущенный процесс (см. рисунок В3.17.9).



**Рисунок В3.17.9** – Характеристики запущенного процесса для монитора

Перед выполнением исследуемой задачи устанавливаем сначала минимальный приоритет процесса (см. рисунок В3.18.10)



**Рисунок В3.17.10** – Понижение базового приоритета

На рисунке В3.17.11 можно наблюдать ход выполнения самой емкой операции при низком приоритете.

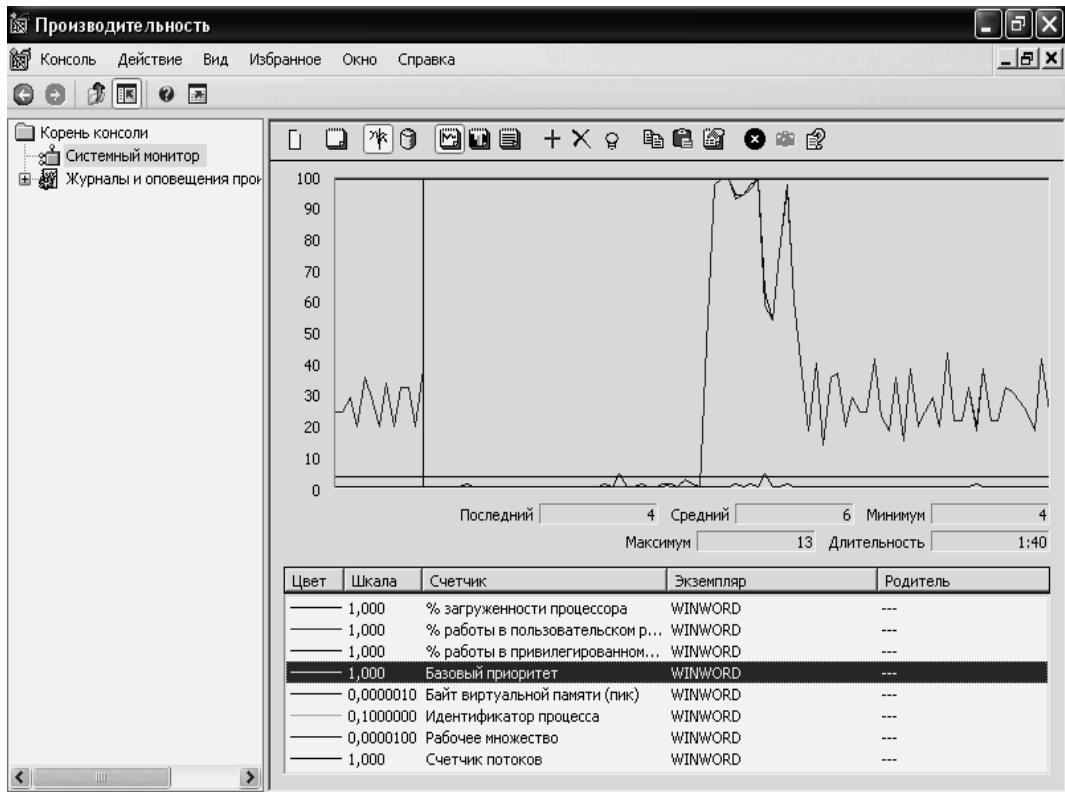


Рисунок В3.17.11 – Выполнение операции при пониженном приоритете

Изменим приоритет запущенного процесса (см. рисунок В3.17.12)

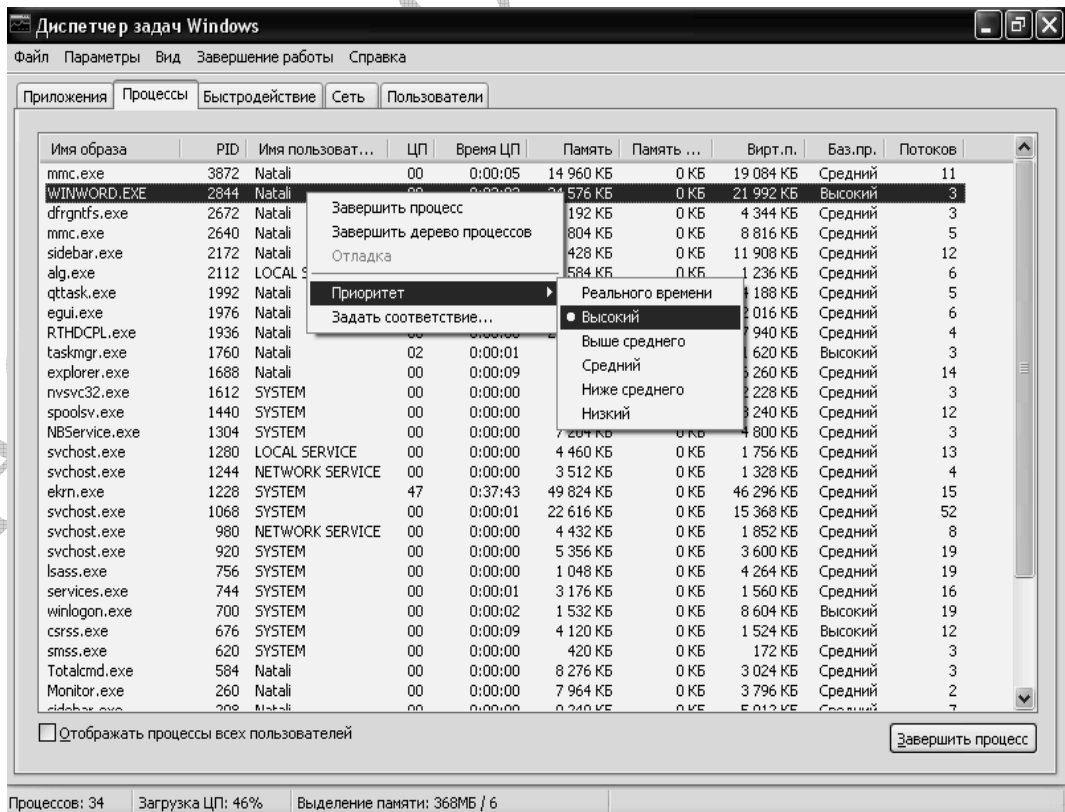
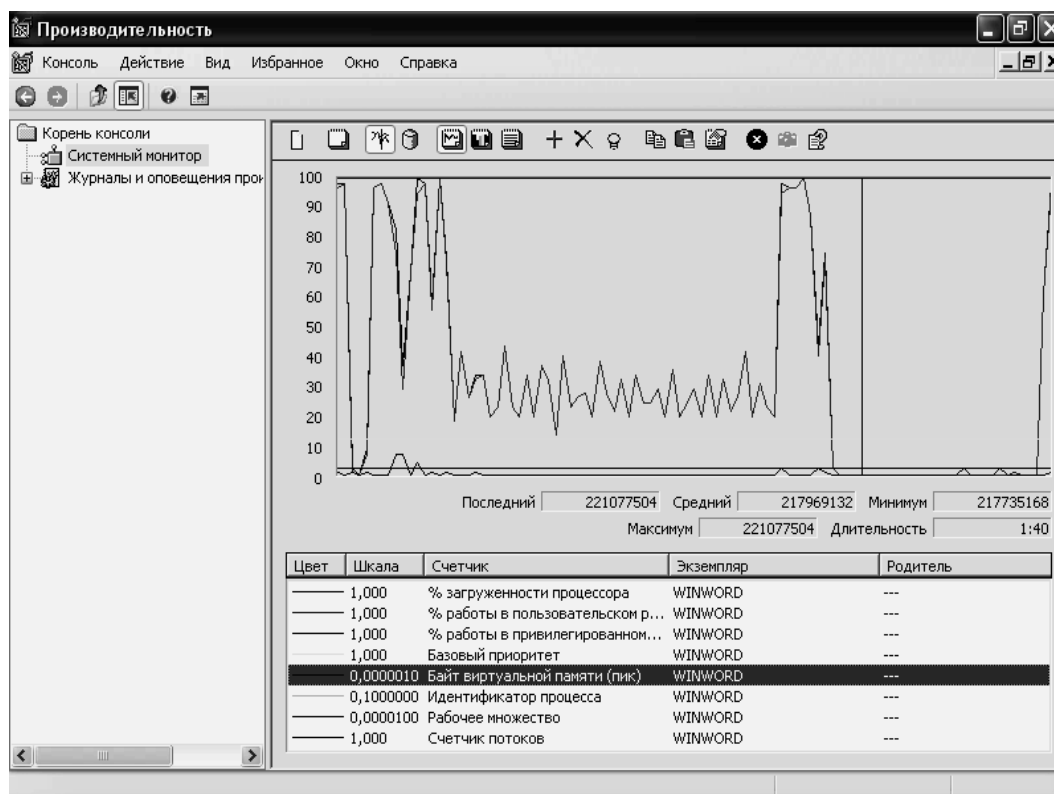


Рисунок В3.17.12 – Повышение базового приоритета

На рисунке В3.17.13 показано выполнение той же операции с повышением приоритета.



**Рисунок В3.17.13** –Процесс выполняется с повышенным приоритетом

Из представленных рисунков очевидно, что при изменении приоритета процесса резко изменяется характер работы процесса в пользовательском режиме в много процессной системе - задача выполняется быстрее.

Диспетчер задач может показать количество потоков, созданных в конкретном процессе. В рассматриваемом процессе было создано 3 потока. Эта многопоточность в рамках одного процесса положительно влияет на производительность задачи.

Однако управлять потоками штатными средствами системы нельзя. Сегодня разработчики программного обеспечения предлагают разнообразные программы для получения более детальной информации о компонентах вычислительного процесса, например Microsoft Spi++ .

## ЛИТЕРАТУРА

- 1 Жадановская Н.П. Операционные системы. Базовый курс [Текст]: учебное пособие для студентов заочной формы обучения. – СПб.: СПбГТИ(ТУ), 2009. –101 с.
- 2 Олифер В.Г., Олифер Н.А. Сетевые операционные системы. Учебник для ВУЗов. - СПб.: ПИТЕР, 2007.- 544с.
- 3 Microsoft Windows XP Home Edition и Professional/ Русские версии/ Под общей редакцией А.Н. Чекмарева. – СПб.: БХВ-петербург, 2006. -640с.
- 4 Дорот В.Л., Новиков Ф.А. Толковый словарь современной компьютерной лексики. – - СПб.: "БХВ - Санкт - Петербург", 2004. - 604с.
- 5 Пройдаков Э.М., Теплицкий Л.А. Англо-русский толковый словарь по вычислительной технике, Интернету и программированию. - М: Русская редакция, 2003. - 634с.
- 6 С. В. Черников, С. Н. Абражевич. Как установить 25 ОС на одном компьютере с нуля! (+ DVD-ROM). Издательство: Лучшие Книги, 2008. - 320с.
- 7 С. Н. Абражевич. Самоучитель. Зоопарк операционных систем на одном компьютере + (DVD-ROM ) Издательство: Лучшие Книги, 2008. - 320с.
- 8 Таненбаум Э. Современные операционные системы. Пер. с англ. - СПб: Питер, 2007 г. 1040с.
- 9 Руссинович М., Соломон Д. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP, Windows 2000. Мастер-класс. СПб:Питер, Русская Редакция, 2005 г. 992 с.
- 10 Windows XP Professional. Учебный курс MCSA/MCSE. Сертификационный экзамен 70-270 (+ CD-ROM)Издательство: Русская Редакция, 2008 г., 704 с.

Кафедра систем автоматизированного проектирования и управления

Методические указания к выполнению контрольных работ  
для студентов заочной формы обучения  
направления подготовки «Информатика и вычислительная техника»

## ОПЕРАЦИОННЫЕ СИСТЕМЫ

Наталья Павловна Жадановская

---

Отпечатано с оригинал макета. Формат 60x90 <sup>1</sup>/<sub>16</sub>  
Печ. л. 7,5. Тираж 100 экз.

---

Государственное образовательное учреждение  
высшего профессионального образования  
Санкт-Петербургский государственный технологический институт  
(технический университет)

---

190013, г. Санкт-Петербург, Московский пр., д. 26