Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
Национальный минерально-сырьевой университет «Горный»

Кафедра информатики и компьютерных технологий

ИНФОРМАТИКА

РЕШЕНИЕ БАЗОВЫХ ГЕОДЕЗИЧЕСКИХ ЗАДАЧ

Методические указания к курсовой работе по информатике для студентов направлений бакалавриата 120700 и 270800 специальности 120401

САНКТ-ПЕТЕРБУРГ
2013

УДК 681.142.2(073)

ИНФОРМАТИКА. РЕШЕНИЕ БАЗОВЫХ ГЕОДЕЗИЧЕСКИХ ЗАДАЧ: Методические указания к курсовой работе / Национальный минерально-сырьевой университет «Горный». Сост.: *В.В.Глазков, А.Б.Маховиков.* СПб, 2013. 52 с.

Рассмотрены основы использования вычислительных возможностей программ общего назначения при решении основных (базовых) геодезических задач. Показаны способы организации расчетов с помощью Microsoft Excel, MathCad, MatLab и системы программирования Visual Basic.

Предложены варианты заданий и указания по выполнению и оформлению курсовой работы.

Методические указания предназначены для студентов направлений бакалавриата 120700 «Землеустройство и кадастры» и 270800 «Строительство» специальности 120401 «Прикладная геодезия».

Табл.17. Ил.7. Библиогр.: 4 назв.

Научный редактор проф. *А.В.Зубов*

Введение

Значительная часть работы геодезиста связана с проведением расчетов. В настоящее время основным вычислительным средством является персональный компьютер. Вычислительные возможности современных компьютеров реализованы как в программных средствах общего назначения, так и специализированном программном обеспечении.

Настоящие методические указания предназначены для организации выполнения студентами курсовой работы по информатике. Описываются основы использования вычислительных возможностей программ общего назначения в проведении геодезических расчетов. Рассказывается о проведении расчетов в Microsoft Excel, MathCad, MatLab и с помощью системы программирования Visual Basic.

Рассматриваются основные (базовые) геодезические расчеты: расчет расстояния до неприступной точки, вычисление превышения, решение обратной геодезической задачи, решение прямой угловой засечки по формулам Гаусса, решение прямой угловой засечки по формулам Юнга, решение обратной угловой засечки по формулам Пранис-Праневича.

Материал изложен так, что сначала приводятся расчетные схемы и формулы, а затем описываются средства и возможности различных программ.

Цель методических указаний – помощь в освоении вычислительных возможностей программ общего назначения при решении геодезических задач. Они предназначены для студентов, специализирующихся в области геодезии.

1. Основные геодезические расчеты

В качестве введения в геодезические расчеты приведем решение одной задачи, предполагающее выполнение вычислительных операций, применяемых при решении геодезических задач.

Пусть треугольник задан координатами своих вершин (рис.1). Вычислим величины его углов. Ответ представим в градусах, минутах, секундах и их десятых долях.

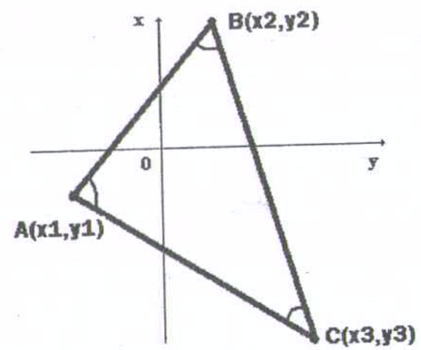

Рис. 1. Треугольник, заданный координатами своих вершин

На рис. 1 система координат выбрана так, как это принято в геодезии.

В случае, когда треугольник задан координатами своих вершин, длины его сторон можно определить по формулам вычисления расстояния между двумя точками, заданными своими координатами. Например, для стороны AB будем иметь

$$AB = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

Длины остальных сторон вычисляются аналогично.

По известным длинам сторон треугольника может быть вычислена его площадь. Для этого можно использовать формулу Герона:

$$S = \sqrt{p(p - AB)(p - BC)(p - AC)},$$

где $p = (AB + BC + AC)/2$ - полупериметр треугольника.

Дальнейшее решение этой задачи может быть основано на использовании теоремы косинусов для треугольника. Согласно этой теореме: квадрат длины одной из сторон треугольника равен сумме квадратов длин двух других сторон без удвоенного произведения этих сторон на косинус угла между ними. Например, для стороны AB будем иметь

$$AB^2 = BC^2 + AC^2 - 2\,BC\,AC\,\cos\hat{C}.$$

Откуда для косинуса угла будем иметь

$$\cos\hat{C} = \frac{BC^2 + AC^2 - AB^2}{2\,BC\,AC}.$$

Для самого угла будем иметь формулу

$$\hat{C} = \arccos\left(\frac{BC^2 + AC^2 - AB^2}{2\,BC\,AC}\right).$$

Аналогично вычисляются остальные два угла треугольника.

Тестовый пример: x1=-3, y1=0, x2=4, y2=-3, x3=8, y3=7;

$\hat{C} = 35°43'38,6''$; $\hat{A} = 55°40'11,2''$; $\hat{B} = 88°36'10,1''$.

К основным (базовым) геодезическим расчетам можно отнести: расчет расстояния до неприступной точки, вычисление превышения, решение обратной геодезической задачи, решение прямой угловой засечки по формулам Гаусса, решение прямой угловой засечки по формулам Юнга, решение обратной угловой засечки по формулам Пранис-Праневича.

Рассмотрим алгоритмы проведения основных геодезических расчетов.

Расчет неприступного расстояния

Пусть необходимо вычислить расстояние от точки A до точки B, между которыми располагается неприступное препятствие (рис. 2).

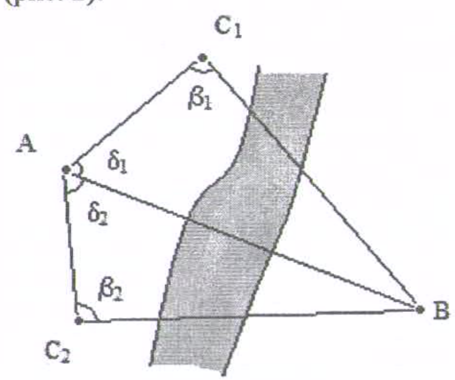

Рис. 2. Схема для вычисления неприступного расстояния

Предположим, что мы располагаем данными измерений, согласно которым известны расстояния AC_1, AC_2 и т.д. (базисы), углы β_1, β_2 и т.д. и углы δ_1, δ_2 и т.д.

Для того, чтобы произвести вычисления запишем теорему синусов для треугольника ABC_1:

$$\frac{AB}{\sin\beta_1} = \frac{AC_1}{\sin(180 - \delta_1 - \beta_1)}.$$

Из этой формулы выразим AB:

$$AB = AC_1\frac{\sin\beta_1}{\sin(180 - \delta_1 - \beta_1)}.$$

Если в нашем распоряжении будет четыре комплекта измерений, то в качестве самого точного значения АВ следует выбрать среднее значение, вычисленное по четырем значениям:

$$AB_{ср} = \frac{(AB)_1 + (AB)_2 + (AB)_3 + (AB)_4}{4}.$$

Расчет для тестового примера:

№	АС	δ		β		АВ	АВ$_{ср}$
		град	мин	град	мин		
1	225,78	81	14	54	37	264,27950	264,29
2	227,38	88	55	49	55	264,28660	
3	193,48	87	25	55	29	264,28097	
4	241,19	89	49	47	43	264,28901	

Расчет превышения

Пусть даны результаты измерений (рис. 3), проведенных с помощью геодезических приборов.

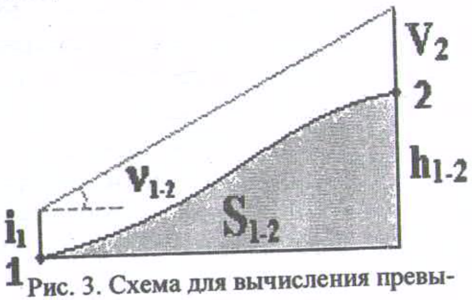

Рис. 3. Схема для вычисления превышения

Для того чтобы определить превышение точки 1 над точкой 2 необходимо с помощью прибора, расположенного на высоте над поверхностью земли i_1, установленного в точке 1, измерить угол ν_{1-2} между горизонтальной плоскостью и направлением на верхний край рейки, длиной V_2, и установленной вертикально в точке 2. Кроме этого в распоряжении вычислителя имеется растояние S_{1-2}. Таким образом, для вычисления h_{1-2} необходимо исходить из известности: расстояния S_{1-2}, угла ν_{1-2}, высот i_1 и V_2. Кроме величины расстояния S_{1-2} выраженного в метрах требуется величина $S_{км}$. Это та же величина, но выраженная в километрах ($S_{км}=S_{1-2}/1000$). Расчет производится по формуле

$$h_{1-2}= S_{1-2}\, tg\, \nu_{1-2} + i_1 - V_2 + 0.0675\, S^2_{км}.$$

Тестовый пример: i_1=1.42 м; S_{1-2}=1785.4 м; ν_{1-2}=-1°17.8'; V_2=3.00 м;

Ответ: h_{1-2}=-41.78 м.

Решение прямой угловой засечки по формулам Юнга

Рассмотрим расчет координат удаленной точки по формулам Юнга. Даны точки 1 и 2 с координатами x_1, y_1 и x_2, y_2, соответственно. Кроме этого с помощью угломерного инструмента измерены два угла β_1 и β_2 (рис. 4).

При решении геодезических задач в отличие от общепринятых вычислений предполагается, что оси координат ох и оу переставлены местами (см. рис.1).

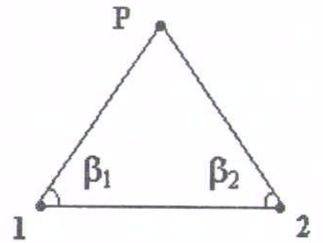

Рис. 4. Расчет координат по формулам Юнга

Расчет координат точки Р можно выполнить по формулам:

$$x_P = \frac{x_1 ctg\beta_2 + x_2 ctg\beta_1 - y_1 + y_2}{ctg\beta_1 + ctg\beta_2}$$

$$y_P = \frac{y_1 ctg\beta_2 + y_2 ctg\beta_1 + x_1 - x_2}{ctg\beta_1 + ctg\beta_2}$$

Тестовый пример:

x1=50; y1=50; x2=30; y2=70; β_1=26°00'09.8"; β_2=58°31'00.7";
xp= 42,112; yp= 72,912.

Решение прямой угловой засечки по формулам Гаусса

Задача ставится так, что требуется вычислить координаты удаленной точки Р по координатам двух заданных точек и дирекционным углам (рис. 5).

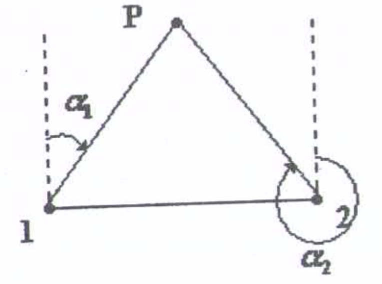

Рис. 5. Схема для вычисления координат по формулам Гаусса

Расчет неизвестных координат точки P может быть произведен по формулам Гаусса:

$$x_p = \frac{x_1 \mathrm{tg}\alpha_1 - x_2 \mathrm{tg}\alpha_2 + y_2 - y_1}{\mathrm{tg}\alpha_1 - \mathrm{tg}\alpha_2}$$

$$y_p = y_1 + (x_p - x_1)\mathrm{tg}\alpha_1 .$$

Тестовый пример: для исходных данных x_1=50; y_1=60; α_1=23°00.2' и x_2=20; y_2=80; α_2=350°07.3' расчет дает x_p=74.68; y_p=70.48.

Решение обратной геодезической задачи

Задача ставится так, что требуется вычислить расстояние и дирекционный угол для двух точек, заданных своими координатами (рис. 6).

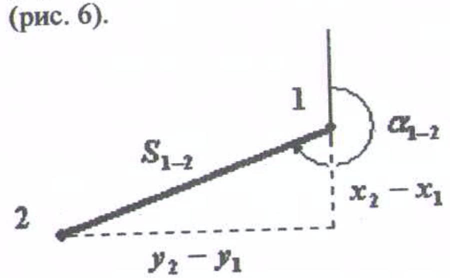

Рис. 6. Схема для решения обратной геодезической задачи

Из рисунка видно, что для тангенса дирекционного угла можно записать соотношение: $\mathrm{tg}\,\alpha_{1-2} = \dfrac{y_2 - y_1}{x_2 - x_1}$.

Значит для угла

$$\alpha_{1-2} = \mathrm{arctg}(\frac{y_2 - y_1}{x_2 - x_1}).$$

При проведении конкретных расчетов следует иметь в виду блок-схему, изображенную на рис. 7. Расстояние может быть вычислено по формуле:

$$S_{1-2} = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2} .$$

Тестовый пример: для исходных данных x_1=6; y_1=5; x_2=1; y_2=3 расчет дает: α_{1-2}=201°48.1', S_{1-2}=5.39.

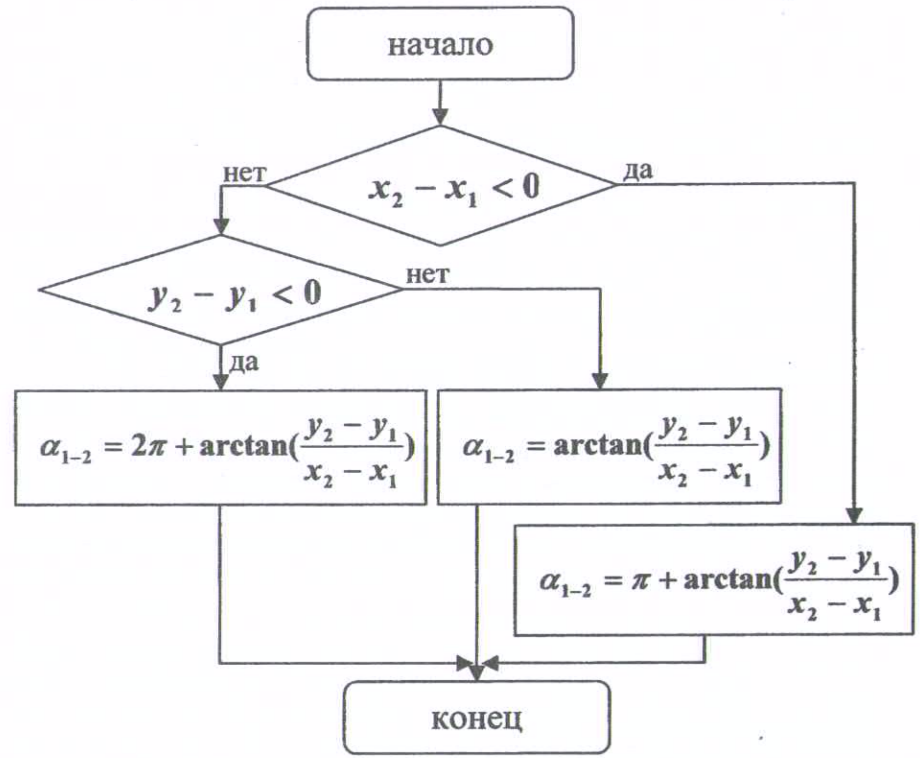

Рис. 7. Схема для выбора формулы для вычисления угла

Решение обратной угловой засечки по формулам Пранис-Праневича

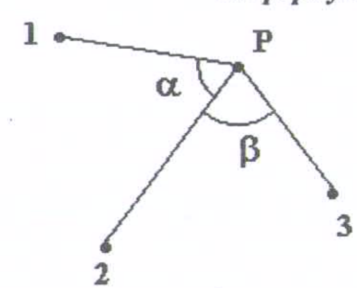

Рис. 8. Схема для вычисления по формулам Пранис-Праневича

Задача ставится так, что требуется вычислить координаты точки P по координатам трех заданных точек и двум углам (рис. 8).

Расчет неизвестных координат точки P может быть произведен по формулам Пранис-Праневича:

$$tgQ = \frac{(y_2 - y_1)ctg\alpha - (y_3 - y_2)ctg\beta + x_1 - x_3}{(x_2 - x_1)ctg\alpha - (x_3 - x_2)ctg\beta - y_1 + y_3};$$

$$N = (y_2 - y_1)(ctg\alpha - tgQ) - (x_2 - x_1)(1 + ctg\alpha\, tgQ);$$

$$\Delta x = \frac{N}{1 + tg^2 Q}; \ \Delta y = \Delta x \ tgQ;$$

$$x_p = x_2 + \Delta x; \qquad y_p = y_2 + \Delta y.$$

Тестовый пример: для исходных данных x_1=3, y_1=1; x_2=1, y_2=4; x_3=2, y_3=7; α=20°15.3', β=19°01.0' расчет дает x_p=10.37; y_p=2.28.

2. Решение вычислительных задач по геодезии с помощью электронных таблиц Microsoft Excel

Для проведения вычислений можно использовать вычислительные возможности программы Microsoft Excel.

Рассмотрим решение тестового примера по расчету углов треугольника с помощью программы MS Excel. Предположим, что координаты вершин треугольника заданы в ячейках: x1→A2, y1→B2, x2→C2, y2→D2, x3→E2, y3→F2 (ячейки первой строки могут быть заполнены поясняющими подписями). Если запланировать ячейку G2, например, под хранение длины стороны AB, то в нее необходимо записать формулу:

=КОРЕНЬ((A2-C2)*(A2-C2)+(B2-D2)*(B2-D2)).

Здесь используется функция вычисления квадратного корня, конструкция (A2-C2)*(A2-C2) соотвествует возведению в квадрат выражения в скобках (можно было бы так же использовать конструкцию (A2-C2)^2). Для сторон BC и AC следует поступить аналогично, выделив, например, под них ячейки H2 и I2.

Для вычисления величины угла A в ячейку J2 необходимо поместить формулу:

=(180/ПИ())*ACOS((G2*G2+I2*I2-H2*H2)/(2*G2*I2)).

Здесь ACOS обозначает вычисление функции арккосинус, G2*G2 – возведение в квадрат, конструкция ПИ() возвращает значение математической константы π, множитель 180/ПИ() используется для превода в градусы величины полученной в радианах. Без этого множителя будет вычислена радианная мера

угла. Для перевода из градусов в радианы необходимо использовать множитель ПИ()/180.

Для вычисления величин углов B и C используем ячейки K2 и L2, поместив в них соотвествующие формулы, аналогичные последней.

Таким образом в ячейках J2, K2 и L2 будем иметь величины искомых углов в градусах и долях. Для проверки расчетов в ячейку M2 следует поместить формулу: =СУММ(J2:L2). Она будет вычислять сумму углов треугольника, которая, как известно, должна быть равна 180 градусам.

Для представления их в градусах – минутах – секундах и долях необходимо выполнить дополнительные вычисления. Отведем для градусов угла A ячейку N2, для минут – O2, для секунд и долей - P2. Для получения градусов необходимо отбросить у величины, хранящейся в ячейке J2 дробную часть. Это можно сделать помещением в N2 формулы: =ОКРВНИЗ(J2;1). Для получения минут в ячейку O2 поместим формулу:
=ОКРВНИЗ((J2-N2)*60;1).
Для получения секунд с долями в P2 поместим формулу:
=(J2-(N2+O2/60))*3600.

Для углов B и C необходимо поступать аналогично, выбрав для них по три ячейки, например, R2, S2, T2 и V2, W2, X2. Расчет дает для углов A, B и C:

Угол A			Угол B			Угол C		
град	мин	сек	град	мин	сек	град	мин	сек
55	40	11,2	88	36	10,1	35	43	38,6

Расчет неприступного расстояния

Рассмотрим решение тестового примера по расчету расстояния до неприступной точки с помощью программы MS Excel.

Спланируем оформление вычислений так, что в ячейках A1-J1 разместим подписи к столбцам. В A2 поместим номер набора данных, которые позволят получить одну величину неприступного расстояния. В ячейке B2 будем задавать базис, в C2 и D2 разместим градусы и минуты угла δ. В ячейку E2 поместим формулу =C2+D2/60. Она будет содержать угол δ в градусах с долями.

Ячейки F2, G2 и H2 будут содержать ту же информацию, но для угла β, т.е. в H2 будет содержаться величина этого угла в градусах с долями (=F2+G2/60). По значениям, содержащимся в B2, E2 и H2 можно вычислить искомую величину неприступного расстояния с помощью формулы, помещенной в I2:

=B2*SIN(H2*ПИ()/180)/SIN((180-E2-H2)*ПИ()/180).

В этой формуле множитель ПИ()/180 служит для перевода градусов с долями в радианы. Таким образом, в ячейке I2 будет находиться величина неприступного расстояния для первого набора данных.

После этого описанные выше аналогичные действия произведем с ячейкам третьей, четвертой и пятой строки, в результате чего в ячейках I2-I5 получим четыре значения неприступного расстояния. Последнее вычислительное действие, которое необходимо выполнить – это помещение в ячейку J2 формулы для вычисления среднего значения: =СУММ(I2:I5)/4. И, наконец, выполним объединение ячеек J2-J5. Для этого выделим с помощью мыши названные ячейки, а затем выполним следующую последовательность действий: пункт главного меню Формат / Ячейки / вкладка Выравнивание / по горизонтали – по центру / по вертикали – по центру / флажок объединение ячеек / кнопка Ок.

Для исходных данных тестового примера получаются следующие результаты:

A	B	C	D	E	F	G	H	I	J
№	AC	δ,г	δ,м	δ	β,г	β,м	β	AB	Среднее
1	225,78	81	14	81,23	54	37	54,62	264,27456	264,29
2	227,38	88	55	88,92	49	55	49,92	264,29168	
3	193,48	87	25	87,42	55	29	55,48	264,28703	
4	241,19	89	49	89,82	47	43	47,72	264,29073	

Расчет превышения в MS Excel

При расчете превышения следует спланировать расчетную таблицу, например, так: В A2 поместим i_1, в B2 – S_{1-2}, в D2 – градусы угла ν_{1-2} , в E2 – минуты с десятыми долями угла ν_{1-2}., в G2 – формулу: =РАДИАНЫ(D2+E2/60) (для перевода угла ν_{1-2} в радианы), в H2 - V_2. Тогда в I2 следует поместить формулу:

=B2*TAN(G2)+A2-H2+0,0675*(B2/1000)^2 (TAN – вычисление функции тангенс). Для тестового примера получаем:

A	B	C	D	E	F	G	H	I
i1	S1-2		ν(г)	ν(м)		ν(p)	V2	h1-2
1,42	1785,4		-1	-17,8		-0,02263	3	-41,78

Расчет координат удаленной точки по формулам Юнга

Рассмотрим решение тестового примера по расчету координат удаленной точки по формулам Юнга с помощью программы MS Excel.

Спланируем вычисления так, что координаты x_1, y_1 и x_2, y_2 поместим в ячейки A2, B2, C2, D2. Если угол β_1 задавать в трех ячейках (градусы, минуты, секунды с долями), например в E2, F2 и G2, то угол в градусах с долями получается использованием формулы в H2: =E2+F2/60+G2/3600. Аналогично, для β_2, используя для ввода ячейки I2, J2, K2, а для формулы L2: =I2+J2/60+K2/3600.

Для расчета координат по формулам Юнга требуется в ячейки A4 и D4 поместить формулы:

=(A2/TAN(РАДИАНЫ(L2))+C2/TAN(РАДИАНЫ(H2))-B2+D2)/(1/TAN(РАДИАНЫ(H2))+1/TAN(РАДИАНЫ(L2)))

и =(B2/TAN(L2*ПИ()/180)+D2/TAN(H2*ПИ()/180)+A2-C2)/(1/TAN(H2*ПИ()/180)+1/TAN(L2*ПИ()/180)).

Здесь TAN – функция, предназначенная для вычисления тангенса угла. Эта функция используется потому что ctgα=1/tgα. Функция TAN должна действовать на аргумент, выраженный в радианах. Это можно сделать либо используя функцию РАДИАНЫ (так сделано в формуле для x_p), либо используя умножение на π/180 (так сделано в формуле для y_p).

Для исходных данных получаются следующие результаты:

A	B	C	D	E	F	G	H	I	J	K	L
x1	y1	x2	y2	β1, г	β1, м	β1, с	β1	β2, г	β2, м	β2, с	β2
50	50	30	70	26	0	9,8	26,0027	58	31	0,7	58,5169
xp			yp								
42,112			72,912								

Решение обратной угловой засечки по формулам Пранис-Праневича

Рассмотрим решение тестового примера по расчету координат удаленной точки по формулам Пранис-Праневича с помощью программы MS Excel.

Спланируем вычисления так, что координаты x_1, y_1, x_2, y_2, x_3, y_3 поместим в ячейки A2 – F2. Если угол α задавать в двух ячейках (градусы и минуты с долями), например в G2 и H2, то угол сначала в градусах с долями, а затем и в радианах получается использованием формулы в K2: =РАДИАНЫ(G2+H2/60). Аналогично, для β, используя для ввода ячейки I2, J2, а для формулы L2: =РАДИАНЫ(I2+J2/60). После этого в ячейках K2 и L2 будут находится значения углов в радианах.

В ячейку M2 поместим формулу для вычисления tgQ:

=((D2-B2)/TAN(K2)-(F2-D2)/TAN(L2)+A2-E2)/((C2-A2)/TAN(K2)-(E2-C2)/TAN(L2)-B2+F2).

В ячейку N2 поместим формулу для вычисления N:

=(D2-B2)*(1/TAN(K2)-M2)-(C2-A2)*(1+M2/TAN(K2)).

В ячейки O2 и P2 поместим формулы для вычисления Δx и Δy:

=N2/(1+M2^2) и =O2*M2,

а в ячейки A4 и C4 формулы для xp и yp: =C2+O2 и =D2+P2.

Для исходных данных тестового примера получаются следующие результаты:

	A	B	C	D	E	F	G	H	I	J	…
1	x1	y1	x2	y2	x3	y3	A,г	A,м	B,г	B,м	…
2	3	1	1	4	2	7	20	15,3	19	1,0	…
	xp		yp								
	10,37		2,28								

Решение обратной геодезической задачи

Рассмотрим решение тестового примера по решению обратной геодезической задачи с помощью программы MS Excel.

Спланируем вычисления так, что координаты x_1, y_1, x_2, y_2 поместим в ячейки A2 – D2.

Для определения дирекционного угла в градусах с долями в ячейку E2 поместим формулу, производящую вычисление в зависимости от условий:

=ЕСЛИ(C2-A2<0;180+(180/ПИ())*ATAN((D2-B2)/(C2-A2)); ЕСЛИ(D2-B2<0;360+(180/ПИ())*ATAN((D2-B2)/(C2-A2)); (180/ПИ())*ATAN((D2-B2)/(C2-A2)))).

В этой формуле используется условная функция. Ее синтаксис:

=ЕСЛИ(Логическое_выражение; Выражение_1; Выражение_2).

Здесь Логическое_выражение может состоять из комбинации отдельных более простых логических выражений. При этом они могут быть объединены логическими связками И и ИЛИ. Например, ИЛИ(A1=3;B1=4;C1=5) - это логическое выражение истинно, когда хотя бы одно из отдельных логических выражений истинно; И(A1=3;B1=4;C1=5) - это логическое выражение истинно, когда все отдельные логические выражения истинны. На место параметров Выражение_1 и Выражение_2 могут быть подставлены, в том числе и условные функции, т.е. имеется возможность создавать вложенные условные функции. Например,

=ЕСЛИ(C6=3;0;ЕСЛИ(C6=4;1800;2700)).

Для выделения из значения угла в градусах с долями градусов и минут в ячейки F2 и G2 поместим формулы: =ОКРВНИЗ(E2;1) и =(E2-F2)*60, соответственно. Для вычисления расстояния в ячейку H2 поместим формулу:

=КОРЕНЬ((D2-B2)^2+(C2-A2)^2).

Для исходных данных тестового примера получаются следующие результаты:

	A	B	C	D	E	F	G	H
1	x1	y1	x2	y2	α	α, г	α, м	S
2	6	5	1	3	201,8014095	201	48,1	5,39

3. Решение вычислительных задач по геодезии с помощью системы для математических расчетов MathCad

Для проведения вычислений можно использовать возможности математического пакета MathCad.

Рассмотрим решение тестового примера для расчета углов треугольника с помощью программы MathCad.

Запустим систему MathCad. Будем исходить из того, что координаты вершин треугольника заданы. Будем задавать их с помощью операции присваивания в рабочем поле окна программы. Для координат первой вершины это можно сделать с помощью конструкций: x1:=-3 y1:=0.

Для набора знака := следует воспользоваться кнопкой $:=$ с панели Арифметика. При этом координатам следует присваивать значения координат точек, например, из тестового примера. Для координат остальных вершин x2, y2, x3, y3 необходимо поступать аналогичным образом.

После ввода координат необходимо выполнить вычисление квадратов сторон и самих сторон. Это можно сделать с помощью конструкций:

$$AB2 := (x2 - x1)^2 + (y2 - y1)^2 \quad AB := \sqrt{AB2}.$$

Здесь операция возведения в квадрат оформляется с использованием кнопки x^2 с панели Арифметика. Для остальных длин сторон и их квадратов необходимо поступать аналогично.

Затем следует вычислить величину угла в граудусах с долями:

$$C := \frac{180}{\pi}\,\mathrm{acos}\left[\frac{(BC2 + AC2 - AB2)}{2 \cdot BC \cdot AC}\right].$$

Здесь acos означает обращение к математической функции аркосинус, множитель 180/pi необходим для перевода из радиан в градусы. Дроби набираются с использованием кнопки $/$, операция умножения - $\times$, скобки- $(\,)$, а число π - π с панели Арифметика.

Для вывода окончательных ответов или промежуточных значений необходимо ввести в рабочем окне имя переменной, в которой хранится нужная величина и вслед за ней ввести знак равенства без двоеточия. Для того, чтобы вывести значение угла в градусах с долями необходимо использовать конструкцию: C=35.727. Причем операция вывода производится с использованием кнопки $=$ с

панели Арифметика (не путать с $:=$). После активизации кнопки $=$ результат появляется на экране автоматически. Для вычисления остальных углов следует поступать аналогично.

Из величин углов в градусах и долях необходимо выделить градусы, минуты и секунды с долями. Для этого следует воспользоваться конструкциями:

$$Cg := \mathrm{trunc}(C) \quad Cm := \mathrm{trunc}[(C - Cg) \cdot 60]$$

$$Cs := \mathrm{round}\left[\left[C - \left(Cg + \frac{Cm}{60}\right)\right] \cdot 3600, 1\right],$$

где Cg – градусы угла, Cm – минуты угла, Cs – секунды угла, trunc – функция взятия целой части числа (отбрасывание дробной части), round – функция округления с точностью до одного знака после запятой. Для остальных углов следует поступать аналогично.

После последних вычислений необходимо вывести результат. Это можно сделать с помощью конструкций:

$$Cg = 35 \quad Cm = 43 \quad Cs = 38.6$$

Таким образом, весь вычисляющий код будет иметь вид:

$x1 := -3 \quad y1 := 0 \quad x2 := 4 \quad y2 := -3 \quad x3 := 8 \quad y3 := 7$

$AB2 := (x2 - x1)^2 + (y2 - y1)^2 \quad AB := \sqrt{AB2}$

$BC2 := (x3 - x2)^2 + (y3 - y2)^2 \quad BC := \sqrt{BC2}$

$AC2 := (x3 - x1)^2 + (y3 - y1)^2 \quad AC := \sqrt{AC2}$

$C := \frac{180}{\pi}\,\mathrm{acos}\left[\frac{(BC2 + AC2 - AB2)}{2 \cdot BC \cdot AC}\right]$

$A := \frac{180}{\pi}\,\mathrm{acos}\left[\frac{(AB2 + AC2 - BC2)}{2 \cdot AB \cdot AC}\right]$

$B := \frac{180}{\pi}\,\mathrm{acos}\left[\frac{(AB2 + BC2 - AC2)}{2 \cdot AB \cdot BC}\right]$

$C = 35.727 \quad Cg := \mathrm{trunc}(C) \quad Cm := \mathrm{trunc}[(C - Cg) \cdot 60]$

$Cs := \mathrm{round}\left[\left[C - \left(Cg + \frac{Cm}{60}\right)\right] \cdot 3600, 1\right]$

$A = 55.67 \quad Ag := \mathrm{trunc}(A) \quad Am := \mathrm{trunc}[(A - Ag) \cdot 60]$

$$As := \mathrm{round}\left[\left[A - \left(Ag + \frac{Am}{60}\right)\right] \cdot 3600, 1\right]$$

$B = 88.603 \quad Bg := \mathrm{trunc}(B) \quad Bm := \mathrm{trunc}[(B - Bg) \cdot 60]$

$$Bs := \mathrm{round}\left[\left[B - \left(Bg + \frac{Bm}{60}\right)\right] \cdot 3600, 1\right]$$

$Cg = 35 \quad Cm = 43 \quad Cs = 38.6 \quad Ag = 55 \quad Am = 40 \quad As = 11.2$

$Bg = 88 \quad Bm = 36 \quad Bs = 10.1$

Видно, что получаются те же значения, которые были получены при проведении расчетов при помощи электронных таблиц.

Набранные конструкции могут быть сохранены в виде файла, с тем, чтобы в дальнейшем при необходимости произвести загрузку и расчеты. Следует отметить, что сохранение документов MathCad может быть осуществлено в файл формата rtf (для Word).

Решение задачи на вычисление превышения в MathCad можно произвести с использованием следующего кода:

$i1 := 1.42 \quad s := 1785.4 \quad v2 := 3$

$$n := -\left(1 + \frac{17.8}{60}\right) \quad h := s \cdot \tan\left(n \cdot \frac{\pi}{180}\right) + i1 - v2 + 0.0675 \cdot \left(\frac{s}{1000}\right)^2$$

$h = -41.777$. Округлив полученное значение, будем иметь h=-41.78.

Видно, что получается то же значение, которое было получено при проведении расчетов при помощи электронных таблиц.

Решение задачи вычисления координат удаленной точки по формулам Юнга можно произвести с использованием следующего кода:

$x1 := 50 \quad y1 := 50 \quad x2 := 30 \quad y2 := 70$

$$b1 := 26 + \frac{0}{60} + \frac{9.8}{3600} \qquad b2 := 58 + \frac{31}{60} + \frac{0.7}{3600}$$

$$b1 := b1 \cdot \frac{\pi}{180} \qquad b2 := b2 \cdot \frac{\pi}{180}$$

$$xp := \frac{\frac{x1}{\tan(b2)} + \frac{x2}{\tan(b1)} - y1 + y2}{\frac{1}{\tan(b1)} + \frac{1}{\tan(b2)}} \qquad yp := \frac{\frac{y1}{\tan(b2)} + \frac{y2}{\tan(b1)} + x1 - x2}{\frac{1}{\tan(b1)} + \frac{1}{\tan(b2)}}$$

$xp = 42.112 \quad yp = 72.912$.

Видно, что получаются те же значения, которые были получены при проведении расчетов при помощи электронных таблиц.

Решение обратной угловой засечки по формулам Пранис-Праневича можно произвести с использованием следующего кода:

$x1 := 3 \quad y1 := 1 \quad x2 := 1 \quad y2 := 4 \quad x3 := 2 \quad y3 := 7$ - задание координат

$al := 20 + \frac{15.3}{60} \quad be := 19 + \frac{1.0}{60}$ - задание углов

$al := al \cdot \frac{\pi}{180} \quad be := be \cdot \frac{\pi}{180}$ - перевод градусов в радианы

$$tgq := \frac{[(y2 - y1) \cdot \cot(al) - (y3 - y2) \cdot \cot(be) + x1 - x3]}{[(x2 - x1) \cdot \cot(al) - (x3 - x2) \cdot \cot(be) - y1 + y3]}$$

$n := (y2 - y1) \cdot (\cot(al) - tgq) - (x2 - x1) \cdot (1 + \cot(al) \cdot tgq)$

$dx := \frac{n}{1 + tgq^2} \quad dy := dx \cdot tgq \quad xp := x2 + dx \quad yp := y2 + dy$

$xp = 10.372 \quad yp = 2.283$ - выдача ответа

Видно, что получаются значения, приведенные в тестовом примере.

Решение задачи на вычисление неприступного расстояния в MathCad можно произвести с использованием следующего кода:

$AC1 := 225.78 \quad AC2 := 227.38 \quad AC3 := 193.48 \quad AC4 := 241.19$ - ввод базисов

$$\delta 1 := \left(81 + \frac{14}{60}\right) \cdot \frac{\pi}{180} \qquad \delta 2 := \left(88 + \frac{55}{60}\right) \cdot \frac{\pi}{180}$$

$\delta 3 := \left(87 + \frac{25}{60}\right) \cdot \frac{\pi}{180} \quad \delta 4 := \left(89 + \frac{49}{60}\right) \cdot \frac{\pi}{180}$ - ввод первого угла и перевод в радианы

$\beta1 := \left(54 + \frac{37}{60}\right) \cdot \frac{\pi}{180}$ $\beta2 := \left(49 + \frac{55}{60}\right) \cdot \frac{\pi}{180}$

$\beta3 := \left(55 + \frac{29}{60}\right) \cdot \frac{\pi}{180}$ $\beta4 := \left(47 + \frac{43}{60}\right) \cdot \frac{\pi}{180}$ - ввод второго угла и перевод в радианы

$AB1 := AC1 \cdot \frac{\sin(\beta1)}{\sin(\pi - \delta1 - \beta1)}$ $AB2 := AC2 \cdot \frac{\sin(\beta2)}{\sin(\pi - \delta2 - \beta2)}$

$AB3 := AC3 \cdot \frac{\sin(\beta3)}{\sin(\pi - \delta3 - \beta3)}$ $AB4 := AC4 \cdot \frac{\sin(\beta4)}{\sin(\pi - \delta4 - \beta4)}$ - расчет расстояний

$AB := \frac{AB1 + AB2 + AB3 + AB4}{4}$ - вычисление ответа

$AB = 264.286$. После округления AB=264.29.

4. Решение вычислительных задач по геодезии с помощью системы для математических расчетов MatLab

Для проведения вычислений можно использовать вычислительные возможности математического пакета MatLab.

Рассмотрим решение тестового примера для расчета углов треугольника с помощью системы Matlab. Для этого выполним следующую последовательность действий:

1. Запустим программу Matlab.

2. Нажав кнопку 🗋 на панели инструментов, откроем окно редактирования программы.

3. Будем набирать программные конструкции.

Самое первое действие в программе должно быть очистка экрана от предшествующих надписей. Это делается оператором **clc**. Затем необходимо прочитать исходные значения координат. Для координат первой вершины .это можно сделать с помощью конструкций:

x1=input('x1->'); y1=input('y1->');,

где комбинация букв, заключенная в апострофы (одиночные кавычки), выступает в роли подсказки при вводе. Для координат остальных вершин x2, y2, x3, y3 необходимо поступать аналогичным образом.

После ввода координат необходимо выполнить вычисление квадратов сторон и самих сторон. Это можно сделать с помощью конструкций:

ab2=(x2-x1)^2+(y2-y1)^2 ; ab=sqrt(ab2);.

Здесь знаками ^2 обозначена операция возведения в квадрат, sqrt – позволяет вычислить квадратный корень.

Для остальных длин сторон и их квадратов необходимо выполнить аналогичные действия.

Затем следует вычислить величину угла в граудусах с долями:

c=(180/pi)*acos((bc2+ac2-ab2)/(2*bc*ac));.

Для вычисления остальных углов следует поступать аналогично. В последнем фрагменте множитель 180/pi необходим для перевода из радиан в градусы, а acos означает обращение к встроенной в Matlab функции.

Из величин углов в градусах с долями необходимо выделить градусы, минуты и секунды с долями. Для этого следует воспользоваться конструкциями:

cg=fix(c); cm=fix((c-cg)*60); csec=(c-(cg+cm/60))*3600;,

где c – величина угла в градусах с долями, cg – градусы угла, cm – минуты угла, csec – секунды угла, fix – функция взятия целой части числа (отбрасывание дробной части). Для остальных углов следует поступать аналогично.

После последних вычислений необходимо выдать результат на экран. Это можно сделать, указав имя переменной, в которой хранится соотвествующая вычисляемая величина, например, cg. Для вывода остальных значений следует поступать аналогично. Кроме такого способа вывода вычисленных значений возможен еще такой, когда после вычисляемого выражения не ставится точка с запятой. Это означает, что соотвествующее вычисленное значение будет выдано на экран.

4. Закончив набор, можно запускать программу на исполнение активизацией кнопки на панели инструментов. При этом следует перейти в окно исполнения программы, в котором будут

появляться конструкции подсказок, запрограммированные в операторах ввода. В соответствующих строках следует производить ввод значений:

```
x1->-3
y1->0
x2->4
```

и т.д.

При запуске программы на исполнение и вводе данных для тестового примера получаем на экране представленный ранее результат. Если в программе обнаружатся ошибки, то необходимо устранить их причину и вновь производить запуск программы на исполнение.

Решение задачи на вычисление превышения в MatLab можно произвести с использованием следующей последовательности операторов:

```
% превышение
clc
i1=input('i1------->');
s =input('s-------->');
v2=input('v2------->');
ng=input('nu(grad)->');
nm=input('nu(min)-->');
n=(ng+nm/60);
h=s*tan(n*pi/180)+i1-v2+0.0675*(s/1000)^2
```

Решение задачи вычисления координат точки по формулам Юнга можно произвести с использованием следующей последовательности операторов:

```
% расчет по формулам Юнга
clc
x1=input('x1->');
y1=input('y1->');
x2=input('x2->');
y2=input('y2->');
b1g=input('b1(grad)->');
b1m=input('b1(min)-->');
b1s=input('b1(sec)-->');
```



```
b2g=input('b2(grad)->');
b2m=input('b2(min)-->');
b2s=input('b2(sec)-->');
b1=b1g+b1m/60+b1s/3600 ; b2=b2g+b2m/60+b2s/3600;
b1=b1*pi/180 ; b2=b2*pi/180;
xp=(x1*cot(b2)+x2*cot(b1)-y1+y2)/(cot(b1)+cot(b2))
yp=(y1*cot(b2)+y2*cot(b1)+x1-x2)/(cot(b1)+cot(b2))
```

Решение задачи вычисления координат точки по формулам Гаусса можно произвести с использованием следующей последовательности операторов:

```
% Расчет по формулам Гаусса
clc
x1 = input('x1->');
y1 = input('y1->');
x2 = input('x2->');
y2 = input('y2->');
a1g = input('a1(grag)->');
a1m = input('a1(min)->');
a2g = input('a2(grag)->');
a2m = input('a2(min)->');
a1= a1g+a1m/60; a2=a2g+a2m/60;
a1= a1 * pi / 180; a2 = a2 * pi / 180;
xp= (x1 * tan(a1) - x2 * tan(a2) + y2 - y1) / (tan(a1) - tan(a2))
yp= y1 + (xp-x1)*tan(a1)
```

Решение обратной угловой засечки по формулам Пранис-Праневича можно произвести с использованием следующей последовательности операторов:

```
% расчет по формулам Пранис-Праневича
clc
x1=input ('x1, y1->');
y1=input ('x1, y1->');
x2=input ('x2, y2->');
y2=input ('x2, y2->');
x3=input ('x3, y3->');
y3=input ('x3, y3->');
alg=input ('al(grad)->');
```

```
alm=input ('al(min)-->');
beg=input ('be(grad)->');
bem=input ('be(min)-->');
al = alg + alm / 60;
be = beg + bem / 60;
al = al * pi / 180; be = be * pi / 180;
tgq=((y2-y1)/tan(al)-(y3-y2)/tan(be)+x1-x3)/((x2-x1)/tan(al)-(x3-x2)/tan(be)-y1+y3);
n = (y2 - y1) * (1/tan(al) - tgq) - (x2 - x1) * (1 + 1/tan(al) * tgq);
dx = n / (1 + tgq ^ 2);   dy = dx * tgq;
xp = x2 + dx
yp = y2 + dy
```

Решение обратной геодезической задачи с помощью MatLab можно произвести с использованием следующей последовательности операторов:

```
% ОГЗ
clc
x1 = input('x1->');
y1 = input('y1->');
x2 = input('x2->');
y2 = input('y2->');
al=atan((y2-y1)/(x2-x1))*180/pi;
if ((x2-x1)<0)
  al=180+al;
elseif ((y2-y1)<0)
   al=360+al
end
s=sqrt((y2-y1)^2+(x2-x1)^2)
alg = fix(al)
alm = ((al - alg) * 60)
```

5. Решение вычислительных задач по геодезии с помощью системы программирования Visual Basic

Visual Basic представляет собой интегрированную систему, предназначенную для набора, редактирования, отладки, компиляции и запуска программ. Кроме того, Visual Basic позволяет сохранять

тексты программ в виде текстовых и самозапускающихся файлов на диске, считывать их из файлов, распечатывать и др.

После запуска Visual Basic на экране появляются окна: главное окно, окно формы, окно свойств и редактора кода, которое частично закрыто окном формы (рис. 9). В главном окне находится главное меню системы Visual Basic, панель инструментов и палитра компонентов. Окно формы (его заголовок Form1) представляет собой заготовку или макет окна разрабатываемого приложения. Окно свойств (его заголовок Properties) позволяет менять свойства объектов проекта. После запуска Visual Basic в окне свойств находятся свойства формы Form1 (как одного из объектов).

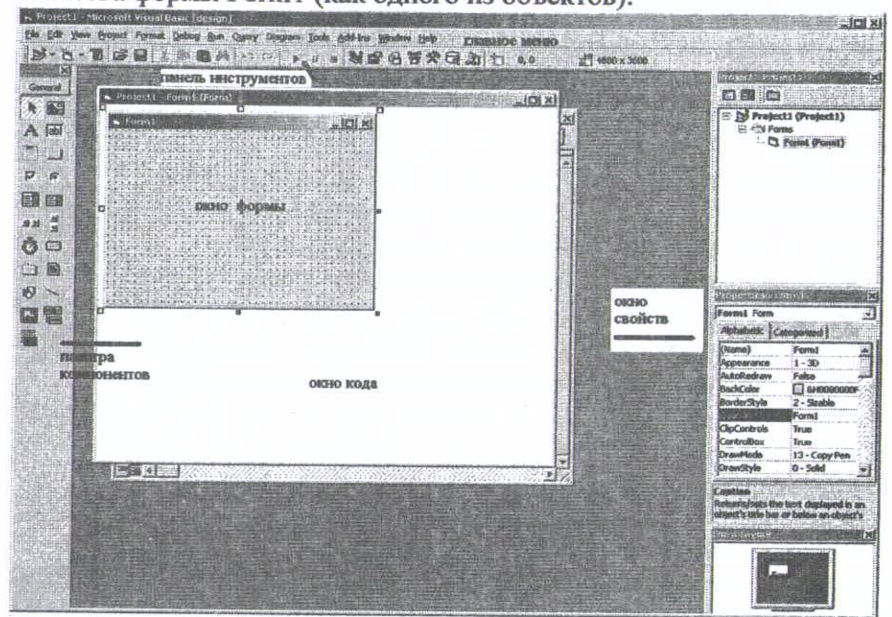

Рис. 9. Вид экрана после запуска Visual Basic

В Visual Basic существует возможность автоматического дополнения кода в программу. Необходимо ввести имя объекта, поставив в конце точку, например: Text1. После некоторой паузы Visual Basic отобразит на экране список всех свойств и методов данного объекта. Из этого списка следует выбрать необходимое свойство и нажать клавишу Enter.

Редактор кода позволяет осуществлять перенос и копирование текста с помощью мыши методом перетаскивания. Чтобы перенести какой-либо фрагмент текста, необходимо отметить (выделить) его, выполнить на нем щелчок мышью и, не отпуская левую кнопку, перетащить фрагмент на нужное место окна редактора кода. При копировании фрагмента следует удерживать нажатой клавишу [Ctrl].

Чтобы сохранить проект, надо из меню File выбрать команду Save Project As. Если проект еще ни разу не был сохранен, то в ответ на команду сохранения проекта Save Project As Visual Basic сначала выводит диалоговое окно Save File As. В начале для сохраняемого проекта необходимо создать собственную папку. Это можно сделать в открытом окне сохранения файла, нажав кнопку . В поле Имя Файла надо ввести имя программного модуля. После щелчка на кнопке Сохранить в диалоговом окне Save File As на диск сохраняется файл с расширением frm и появляется следующее диалоговое окно - Save Project1 As. В поле Имя файла следует ввести имя проекта и нажать на кнопку сохранить. При этом будет сохранен файл с раширением vbp. После внесения изменений в проект, его текущее состояние сохранятся нажатием на кнопку с изображением дискеты на панели инструментов. Рекомендуется первое сохранение проекта делать, когда только начали его разработку, а затем как можно чаще делать сохранение, нажимая кнопку с изображением дискеты.

В терминологии Visual Basic проект - это набор файлов, используя которые компилятор создает файл исполняемой программы. Один из файлов, который называется файлом проекта и имеет расширение vbp, содержит общее описание проекта.

Запуск приложения можно выполнить непосредственно из среды программирования. Для этого надо выбрать команду Start из пункта главного меню Run или нажать кнопку на панели инструментов. Окно приложения - это обычное окно Windows. Его можно перемещать по экрану, можно развернуть на весь экран, можно свернуть и представить в виде пиктограммы.

Visual Basic – среда визуально-ориентированного программирования (ВОП). Согласно принципам технологии ВОП работа над новым проектом начинается с создания стартовой формы окна, которое появляется при запуске приложения. Стартовая форма создается путем изменения свойств формы. Свойства формы определяют ее внешний вид: размер, положение на экране, текст заголовка, вид рамки и т.д. Свойства перечислены в окне свойств (Properties). В левой колонке находятся имена свойств, а в правой - их значения. Сначала надо изменить значение свойства Caption (заголовок) заменить Form1, например, на «Название формы». Для этого следует щелкнуть мышью в поле Caption. В результате этого в правой колонке, где находится текущее значение свойства (фраза Form1), появляется курсор. Используя клавишу <Backspace>, необходимо удалить надпись Form1 и ввести нужное название. Аналогичным образом можно установить значения свойств Height и Width, которые определяют высоту и ширину формы. При выборе некоторых свойств, например Backcolor, которое определяет цвет фона формы, после значения свойств выводится значок выпадающего списка допустимых значений свойства, из которого можно выбрать нужное значение. Помимо обычных свойств у объектов, в том числе и у формы, могут быть вложенные свойства. Например, в поле значения свойства Font расположена кнопка с тремя точками, при нажатии на которую появляется стандартное диалоговое окно Windows выбора шрифта и его свойств. Большинство свойств определяют внешний вид формы. Свойство (Name) определяет имя формы, оно используется в программе для управления формой.

Для управления проектом, представления информации в нужном виде, совершения действий над его объектами служат компоненты формы, расположенные на палитре компонентов. Чтобы добавить к форме компонент, надо в палитре компонентов щелкнуть на пиктограмме нужного компонента, и затем кнопкой мыши сделать прорисовку объекта в том месте формы, где должен находиться выбранный компонент. В результате в форме появляется нужный компонент. Компонент формы, окруженный восемью маленькими квадратиками, считается выделенным (маркированным). Свойства маркированного компонента отображается в окне свойств. Visual Basic позволяет легко изменить положение и размер компонента. Чтобы изменить положение компонента, надо установить курсор мыши на изображение компонента, нажать левую кнопку мыши и, удерживая ее нажатой, переместить изображение компонента в нужную точку формы. Затем отпустить кнопку мыши. Так же, как свойства компонента можно изменить в окне свойств. Чтобы свойства требуемого компонента появились в данном окне, надо маркировать нужный компонент или выбрать его имя из раскрывающегося списка объектов, кнопка раскрытия которого находится в верхней части окна свойств.

Visual Basic – среда событийно-ориентированного программирования. Внутренняя структура программ для Windows кардинальным образом отличается от структуры консольных программ, где операторы выполнялись последовательно, от начала программы до ее окончания. Операционная система Windows и программы, работающие в ней, функционируют по другому принципу. Они обрабатывают возникающие в них события: щелчок мыши на кнопке, выбор пункта меню, нажатие клавиши, достижение встроенным таймером заданного значения времени - и передают их выполняющимся в своей среде программам (подпрограммам). Они, в свою очередь, обычно находятся в состоянии ожидания и активизируются только при получении от Windows сообщений о событиях - реагируют на них. Конечно, некоторые программы могут выполнять длительные вычисления, например в фоновом режиме, но это нетипично. Сообщения Windows обрабатываются программой не одновременно, а последовательно (хотя некоторые сообщения имеют более высокие приоритеты, чем другие). Например, сообщение, требующее завершения работы приложения, не выполняется мгновенно, а ставится в очередь сообщений, и, пока не будут обработаны первые сообщения (это может быть информация о щелчках мыши, нажатиях клавиш, кнопок и прочее), работа программы не продолжится. Структура программы для Windows представляет собой набор подпрограмм, каждая из которых ответственна за обработку конкретного события и вызывается только при его получении. Программист сам решает, какие события в программе требуется обрабатывать. В нашем примере необходимо реагировать только на щелчок по кнопке. Системные события: выбор пункта системного меню, закрытие приложения - обрабатываются в программе, созданной с помощью системы Visual Basic, автоматически.

Подобный подход к созданию программ называется событийно - ориентированным. Реакцией на событие должно быть какое-либо действие. В Visual Basic реакция на событие реализуется как процедура обработки события. Таким образом, задача программиста состоит в написании необходимых процедур обработки событий. Для создания подобных процедур сначала следует выделить объект, для которого создается процедура обработки события, а затем выполнить двойной щелчок мышью. В результате открывается окно редактора кода с макетом процедуры обработки события. Макет процедуры – это две строки, символизирующие начало и конец процедуры. Например,

```
Private Sub Command1_Click()
End Sub
```

Visual Basic автоматически присваивает процедуре обработки события имя. Имя процедуры обработки события состоит из двух частей, разделенных знаком подчеркивания «_». Первая часть идентифицирует сам объект, вторая событие. Например, Command1_Click. В окне редактора кода между строками каркаса процедуры можно печатать инструкции входного языка Visual Basic, реализующие процедуру обработки события.

Доступ к свойствам объектов происходит точно так же, как к полям записей. Сначала указывается имя переменной, определяющей «владельца» этих свойств, а затем через точку приводятся названия свойств: Text1.Text, Label1.Caption и др.

Если процедура обработки события больше не нужна, ее можно удалить, удалив весь программный код, включая комментарии внутри ее. Пустая процедура при последующей компиляции будет автоматически удалена.

Ввод значений переменных может быть осуществлен с помощью компонента Text и некоторых встроенных процедур. При этом для a, являющейся строкой, используется конструкция: a=Text1.Text, для числового b : b=Val(Text1.Text). Здесь Val – это конструкция языка программирования Бейсик, преобразующая значение текстовой переменной в числовой тип.

Вывод значений переменных может быть осуществлен с использованием компонента Label и некоторых встроенных процедур.

При этом для a являющейся строкой используется конструкция: Label1.Caption=a, для b числового: Label1.Caption=Str(b). Для вывода с подсказкой, например, для вещественной переменной: Label1.Caption = "Подсказка "+Str(a). Здесь Str – это конструкция языка программирования Бейсик, преобразующая значение числовой переменной в строковый тип.

Кроме этой возможности в Visual Basic можно пользоваться для вывода обычным оператором print.

Рассмотрим решение тестового примера по расчету углов треугольника с помощью Visual Basic. Будем исходить из того, что координаты вершин треугольника заданы. Будем помещать их в поля в окне программы.

Запустим систему Visual Basic. Через окно свойств изменим значение свойства Caption для формы со значения Form1 на Треугольник. Поместим на форму одну командную кнопку. Через окно свойств зададим значение свойства Caption равным Расчет. Поместим на форму шесть компонентов Text, разместив около каждого из них по компоненту Label (рис. 10). Через окно свойств свойство Text каждого компонента Text заполним соотвествующим значением координат точек (например, из тестового примера).

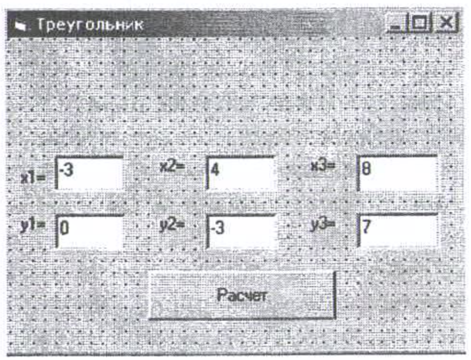

Рис. 10. Вид формы для решения задачи о треугольнике

Если значениями координат являются дробные числа, то в качестве разделителя между целой и дробной частью следует использовать точку.

Подписи полей ввода созданы с помощью компонентов Label, а именно в свойство Caption помещены: x1=, y1= и т.д. Таким образом каждая пара вертикально расположенных полей позволит задавать координаты одной из вершин треугольника.

Выполним два быстрых щелчка на кнопке Расчет. Тем самым будет создан каркас процедуры обработки события, возникающего при нажатии на кнопку:

```
Private Sub Command1_Click()
End Sub
```

Между этими двумя строками необходимо поместить операторы, производящие расчет углов треугольника.

Для осуществления математических вычислений в языках программирования предусмотрены стандартные математические функции. Для большинства версий Бейсика они могут быть представлены таблицей:

Математическое значение	Функция
$\sqrt{x}$	sqr(x)
e^x	exp(x)
2^x	exp2(x)
10^x	exp10(x)
$\lvert x \rvert$	abs(x)
ln x	log(x)
lg x	log10(x)
$\log_2 x$	log2(x)
sin x	sin(x)
cos x	cos(x)
tg x	tan(x)
arctan x	atn(x)
x^y	x^y
x^2	x^2
Датчик псевдослучайных чисел в (0,1)	rnd
Остаток от деления x на y	x mod y
[x] –целая часть	fix(x)
Взятие целого слева от x	int(x)
Знак числового выражения $\begin{cases} 1, если\, x > 0; \\ 0, если\, x = 0; \\ -1, если\, x < 0 \end{cases}$	sgn(x)

В случае, если требуемая функция отсутствует в списке стандартных, прибегают к ее выражению через имеющиеся. Например, для функции арккосинус имеем:

$$\arccos x = A,\ \cos A = x,\ \operatorname{tg} A = \frac{\sin A}{\cos A},\ \sin A = (1-\cos^2 A)^{1/2},$$

$$\operatorname{tg} A = \frac{(1-\cos^2 A)^{1/2}}{\cos A},\ \operatorname{tg} A = \frac{(1-x^2)^{1/2}}{x},\ \operatorname{arctg}\frac{(1-x^2)^{1/2}}{x} = A$$

Окончательно, $\arccos\ x = \operatorname{arctg}\dfrac{(1-x^2)^{1/2}}{x}$. Таким образом, отсутствующая функция сведена к функции, которая есть в таблице стандартных функций.

В Бейсике можно самому определить любую функцию, к которой в дальнейшем можно обратиться как к стандартной. Под функцией пользователя мы будем понимать вспомогательный алгоритм, преобразующий некоторые входные параметры в одно значение, которое возвращается в точку вызова. Схематически функцию можно изобразить как на рис. 11.

Функция определяется следующей структурой:

```
Function имя (список параметров)
   описание переменных
   операторы
   имя = выражение
END Function
```

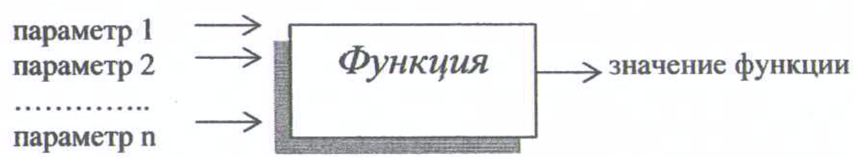

Рис. 11. Схема действия функции

Обращение к функции записывается там, где нужно получить ее значение и осуществляется указанием имени:

имя (список фактических значений параметров)

При этом параметры, используемые при описании, заменяются их фактическими значениями, которые используются для расчетов значений функции.

Для изменения естественного порядка выполнения операторов после проверки некоторого условия предназначен условный оператор. Он предписывает выполнять некоторое действие только в том случае, когда выполняется заданное условие. Это условие записывается в виде логического выражения, а действие, которое нужно выполнить, задается с помощью обычных операторов. Имеются две конструкции организации условных алгоритмов: простая и расширенная. Графически их изображают, как на рис. 12.

Реализация этих конструкций осуществляется при помощи условного оператора IF. Простая конструкция реализуется с помощью программной конструкции:

IF <условие> THEN <действие1>,

а расширенная конструкция:

IF <условие> THEN <действие1> ELSE <действие2> .

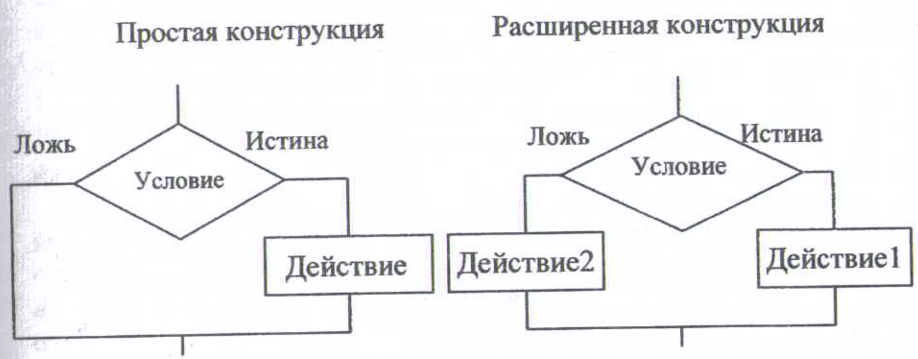

Рис. 12. Схемы алгоритмической структуры "Условный оператор"

С помощью условного оператора можно разветвить программу. Простую конструкцию IF-THEN следует читать так: если условие выполняется (логическое выражение принимает значение "Истина"), то действие, заданное оператором или множеством операторов, выполняется. В противном случае (логическое выражение принимает значение "Ложь") действие не выполняется. В программе, если условие не выполняется, управление передается следующей строке. Конструкция IF-THEN-ELSE дает возможность выбирать одно из двух действий.

<условие> в программе представляет собой логическое выражение, состоящее из двух операндов, соединенных между собой одной из операций отношения: < - меньше, > - больше, <= (или =<) – меньше или равно, >= (или =>) – больше или равно, = - равно, <> (или ><) – не равно.

При расчетах углов треугольника по теореме косинусов требуется вычислять функцию арккосинус. В Бейсике нет стандартной математической функции арккосинус. В этом случае можно воспользоваться выражением функции арккосинус через арктангенс. Прежде чем, запрограммировать в процедуре обработки события расчет углов треугольника, поместим перед описанием этой процедуры функцию:

```
Function acos(x)
  pi = 3.1415926
  aa= Atn(Sqr(1 - x * x) / x)
  if aa<=0 then aa=aa+pi
  if x=-1 then aa=pi
  acos =aa
End Function
```

Эту функцию будем теперь использовать в процедуре обработки события.

Самое первое действие в процедуре обработки события должно быть очистка формы от предшествующих надписей. Это делается оператором cls. Затем необходимо прочитать введенные в поля значения координат. Для координат первой вершины это можно сделать с помощью конструкций:

```
x1 = Val(Text1.Text) : y1 = Val(Text2.Text).
```

Двоеточие между двумя конструкциями выполняет роль разделителя между операторами, когда они помещены в одну строку. Для координат остальных вершин x2, y2, x3, y3 необходимо поступать аналогичным образом.

После ввода координат необходимо выполнить вычисление квадратов сторон и самих сторон. Это можно сделать с помощью конструкций:

```
ab2 = (x2 - x1) ^ 2 + (y2 - y1) ^ 2 : ab = Sqr(ab2).
```

Здесь знаком ^2 обозначена операция возведения в квадрат, Sqr – вычисляет квадратный корень.

Для остальных длин сторон и их квадратов необходимо поступать аналогично.

Затем введя постоянную величину pi=3.1415926, следует вычислить величину угла в граудусах с долями:

```
c = (180 / pi) * acos((bc2 + ac2 - ab2) / (2 * bc * ac)).
```

Здесь acos означает обращение к функции, описанной в программе, множитель 180/pi необходим для перевода из радиан в градусы. Для вычисления остальных углов следует поступать аналогично.

Из величин углов в градусах и долях необходимо выделить градусы, минуты и секунды с долями. Для этого следует воспользоваться конструкциями:

```
cg = Int(c) : cm = Int((c - cg) * 60) : csec = (c - (cg + cm / 60)) * 3600,
```

где c – величина угла в градусах с долями, cg – градусы угла, cm – минуты угла, csec – секунды угла, Int – функция взятия целой части числа (отбрасывание дробной части). Для остальных углов следует поступать аналогично.

После последних вычислений необходимо выдать результат на форму. Это можно сделать с помощью оператора:

```
Print "c="; cg; "grad"; cm; "min"; csec; "sec".
```

Для вывода остальных значений углов следует поступать аналогично.

Таким образом, вся процедура будет иметь вид:

```
Private Sub Command1_Click()
  Cls
  x1 = Val(Text1.Text) : y1 = Val(Text2.Text)
  x2 = Val(Text3.Text) : y2 = Val(Text4.Text)
  x3 = Val(Text5.Text) : y3 = Val(Text6.Text)
  ab2 = (x2 - x1) ^ 2 + (y2 - y1) ^ 2 : ab = Sqr(ab2)
  bc2 = (x3 - x2) ^ 2 + (y3 - y2) ^ 2 : bc = Sqr(bc2)
  ac2 = (x3 - x1) ^ 2 + (y3 - y1) ^ 2 : ac = Sqr(ac2)
```

```
    pi = 3.1415926
    c = (180/pi) * acos((bc2 + ac2 - ab2) / (2 * bc * ac))
    a = (180/pi) * acos((ab2 + ac2 - bc2) / (2 * ab * ac))
    b = (180/pi) * acos((ab2 + bc2 - ac2) / (2 * ab * bc))
    cg = Int(c): cm = Int((c - cg) * 60) : csec = (c - (cg + cm / 60)) * 3600
    ag = Int(a): am = Int((a - ag) * 60) : asec = (a - (ag + am / 60)) * 3600
    bg = Int(b): bm = Int((b - bg) * 60) : bsec = (b - (bg + bm / 60)) * 3600
    Print "c="; cg; "grad"; cm; "min"; csec; "sec"
    Print "a="; ag; "grad"; am; "min"; asec; "sec"
    Print "b="; bg; "grad"; bm; "min"; bsec; "sec"
End Sub
```

В результате вычислений для тестового примера получаем результат, представленный на рис. 13. Видно, что получаются те же значения, которые были получены при проведении расчетов во всех программных системах.

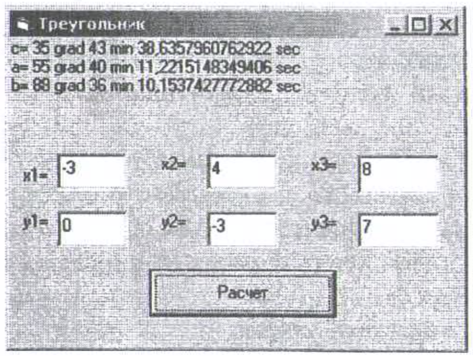

Рис. 13. Вид окна программы для решения задачи о треугольнике

Рассмотрим вычисление превышения с помощью Visual Basic. Будем исходить из того, что входные величины будут заданы в полях окна программы.

Запустим систему Visual Basic. С помощью окна свойств изменим значение свойства Caption для формы со значения Form1 на Превышение. Поместим на форму одну командную кнопку. Пользуясь окном свойств зададим значение ее свойства Caption равным Расчет. Поместим на форму пять компонентов Text, разместив около каждого из них по компоненту Label (рис. 14). Через окно свойств свойство Text каждого компонента Text заполним соотвествующими значениями величин (например, из тестового примера, разделитель между целой и дробной частью - точка).

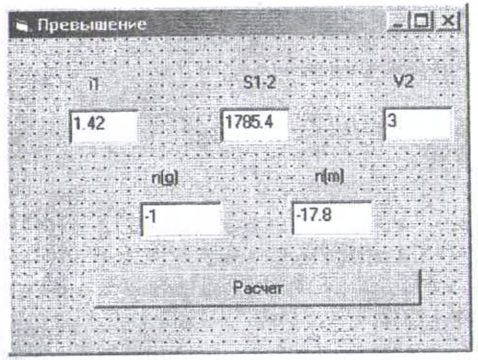

Рис. 14. Вид формы для решения задачи о превышении

Подписи полей ввода созданы с помощью компонентов Label, а именно в свойство Caption помещено: i1, S1-2 и т.д.

Выполним два быстрых щелчка на кнопке Расчет. Тем самым будет создан каркас процедуры обработки события, возникающего при нажатии на кнопку при последующем исполнении программы:

```
Private S  ub Command1_Click()
End Sub
```

Между этими двумя строками необходимо поместить операторы, производящие расчет превышения.

Самым первым действием в процедуре обработки события должно быть очистка формы от предшествующих надписей. Это делается оператором cls. Затем необходимо прочитать введенные в поля значения величин. Для первой величины это можно сделать с помощью конструкции:

i1 = Val(Text1.Text).

Для величин S и V2 нужно поступать аналогичным образом.

Величина n может быть введена следующим образом:

n = Val(Text4.Text) + Val(Text5.Text) / 60,

т.е. значение величины прочитывается с двух полей ввода.

После ввода исходных величин и задав постоянную величину pi = 3.1415926 необходимо выполнить вычисление превышения. Это можно сделать с помощью конструкции:

h = s * Tan(n * pi / 180) + i1 - v2 + 0.0675 * (s / 1000) ^ 2.

Здесь знаком ^2 обозначена операция возведения в квадрат, множитель pi/180 служит для преобразования из градусов в радианы, Tan – обозначает вызов стандартной функции тангенс.

После вычислений необходимо выдать результат на форму. Это можно сделать с помощью оператора:

```
Print "h= "; h.
```

Таким образом, вся процедура будет иметь вид:

```
Private Sub Command1_Click()
    Cls
    pi = 3.1415926
    i1 = Val(Text1.Text): s = Val(Text2.Text)
    v2 = Val(Text3.Text)
    n = Val(Text4.Text) + Val(Text5.Text) / 60
    h = s * Tan(n * pi / 180) + i1 - v2 + 0.0675 * (s / 1000) ^ 2
    Print "h= "; h
End Sub
```

В результате вычислений для тестового примера получаем результат, представленный на рис. 15.

Видно, что получаются то же значение, которое было получено при проведении расчетов при помощи электронных таблиц, MathCad и MatLab.

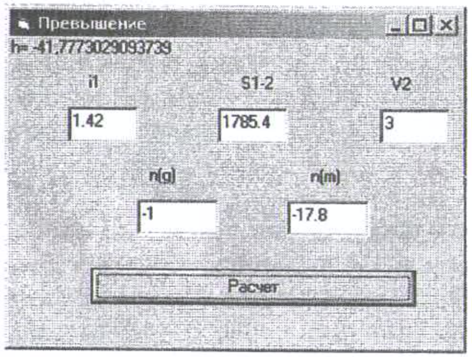

Рис. 15. Вид окна программы для решения задачи о превышении

Рассмотрим решение тестового примера по расчету координат удаленной точки по формулам Юнга с помощью Visual Basic. Будем исходить из того, что координаты двух вершин и два угла треугольника заданы. Будем помещать их в поля окна программы.

Запустим систему Visual Basic. Через окно свойств изменим значение свойства Caption для формы со значения Form1 на Формулы Юнга. Поместим на форму одну командную кнопку. Через окно свойств зададим значение свойства Caption равным Расчет. Поместим на форму десять компонентов Text, разместив около каждого из них по компоненту Label (рис. 16). Через окно свойств свойство Text каждого компонента Text

заполним соотвествующим значением координат точек и величин углов (например, из тестового примера, разделитель между целой и дробной частью - точка).

Подписи полей ввода созданы с помощью компонентов Label, а именно в свойство Caption поместим: x1, y1 и т.д.

Выполним два быстрых щелчка на кнопке Расчет. Тем самым будет создан каркас процедуры обработки события, возникающего при нажатии на кнопку:

```
Private Sub Command1_Click()
End Sub
```

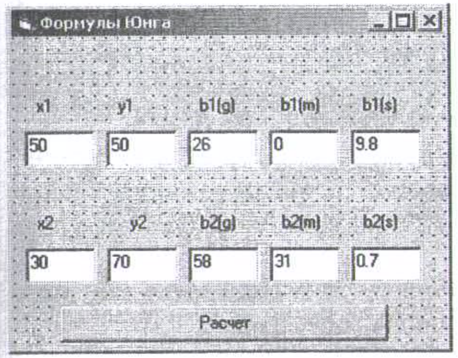

Рис. 16. Вид формы для решения задачи о вычислении координат по формулам Юнга

Между этими двумя строками необходимо поместить операторы, производящие расчет координат по формулам Юнга. При расчетах требуется вычислять функцию котангенс. В Бесике нет стандартной математической функции котангенс. В этом случае можно воспользоваться выражением этой функции через косинус и синус. Прежде чем, запрограммировать расчет в процедуре обработки события, поместим перед описанием этой процедуры функцию:

```
Function ctg(x)
    ctg = Cos(x) / Sin(x)
End Function
```

Эту функцию будем теперь использовать в процедуре обработки события.

Самое первое действие в процедуре обработки события должно быть очистка формы от предшествующих надписей. Это делается оператором cls. Затем необходимо прочитать введенные в поля значения координат и углов. Для координат первой вершины это можно сделать с помощью конструкций:

x1 = Val(Text1.Text) : y1 = Val(Text3.Text)

Двоеточие между двумя конструкциями выполняет роль разделителя между операторами, когда они помещены в одну строку. Для координат второй вершины **x2, y2** и других величин нужно поступать аналогичным образом.

Значение одного угла в градусах, минутах и секундах с долями может быть введено с помощью оператора:

b1 = Val(Text5.Text) + Val(Text7.Text) / 60 + Val(Text9.Text) / 3600.

Для ввода второго угла следует поступать аналогично.

После ввода координат и углов, введя постоянную величину pi = 3.1415926 необходимо выполнить перевод углов из градусов в радианы:

b1 = b1 * pi / 180 : b2 = b2 * pi / 180.

Затем следует вычислить величину координат:

xp = (x1 * ctg(b2) + x2 * ctg(b1) - y1 + y2) / (ctg(b1) + ctg(b2))

Здесь **ctg** означает обращение к функции, описанной в программе. Для вычисления второй координаты следует поступать аналогично.

После последних вычислений необходимо выдать результат на форму. Это можно сделать с помощью оператора:

Print "xp= "; xp, "yp= "; yp.

Таким образом, вся процедура будет иметь вид:

```
Private Sub Command1_Click()
   Cls
   X1 = Val(Text1.Text): Y1 = Val(Text3.Text)
   X2 = Val(Text2.Text): Y2 = Val(Text4.Text)
   b1 = Val(Text5.Text) + Val(Text7.Text) / 60 + Val(Text9.Text) / 3600
   b2= Val(Text6.Text) + Val(Text8.Text) / 60 + Val(Text10.Text) / 3600
   pi = 3.1415926 :  b1 = b1 * pi / 180: b2 = b2 * pi / 180
   xp = (X1 * ctg(b2) + X2 * ctg(b1) - Y1 + Y2) / (ctg(b1) + ctg(b2))
   yp = (Y1 * ctg(b2) + Y2 * ctg(b1) + X1 - X2) / (ctg(b1) + ctg(b2))
   Print "xp= "; xp, "yp= "; yp
End Sub
```

В результате вычислений для тестового примера получаем результат, представленный на рис. 17.

Видно, что получаются те же значения, которые были получены при проведении расчетов при помощи электронных таблиц, MathCad и MatLab.

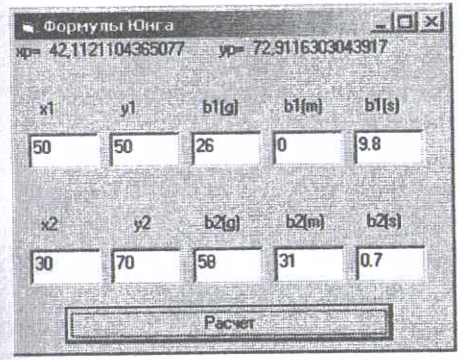

Рис. 17. Вид окна программы для решения задачи о расчете координат по формулам Юнга

Решение задачи вычисления координат удаленной точки по формулам Пранис-Праневича можно произвести с использованием следующей процедуры:

```
Private Sub Command1_Click()
    Cls
    X1 = Val(Text1.Text)
    Y1 = Val(Text2.Text)
    X2 = Val(Text3.Text)
    Y2 = Val(Text4.Text)
    x3 = Val(Text5.Text)
    y3 = Val(Text6.Text)
    al = Val(Text7.Text) +
   Val(Text8.Text) / 60
  be = Val(Text9.Text) + Val(Text10.Text) / 60
  pi = 4 * Atn(1)
  al = al * pi / 180: be = be * pi / 180
  tgq=((y2-y1)/tan(al)-(y3-y2)/tan(be)+x1-x3)/((x2-x1)/tan(al)-(x3-x2)/tan(be)-y1+y3)
  n = (Y2 - Y1) * (ctg(al) - tgq) - (X2 - X1) * (1 + ctg(al) * tgq)
  dx = n / (1 + tgq ^ 2):   dy = dx * tgq
  xp = X2 + dx: yp = Y2 + dy
  Print "xp="; xp, "yp="; yp
End Sub
```

Для организации формы для ввода данных следует использовать приемы, описанные при решении предыдущих задач.

6. Порядок выполнения и представления курсовой работы

Получить задание на курсовую работу следует не позднее, чем через две недели после начала занятий. Во время выдачи заданий объявляются сроки выполнения студентом отдельных этапов, назначается дата сдачи отчёта на проверку и дата защиты работы.

Выполнение курсовой работы предполагает соблюдение следующих этапов:

1. Согласование темы с руководителем работы (срок: первый месяц семестра).
2. Изучение литературы по теме:
   - повторение, углубленное изучение разделов учебников (конспектов лекций), относящихся к выбранной теме;
   - ознакомление с литературой, рекомендуемой настоящими методическими указаниями, конспектирование и цитирование необходимых для решения поставленной задачи теоретических положений;
   - самостоятельный подбор дополнительной литературы;
   - подбор справочного материала.
3. Составление текста заданий в соответствии с номером варианта.
4. Формализация исходной информации, выбор методов решения.
5. Алгоритмизация и написание текстов программ, их перенос на компьютер.
6. Отладка программ, получение окончательных результатов.
7. Написание пояснительной записки (отчета о работе).
8. Подготовка графического материала, иллюстрирующего содержание работы и выводы автора.
9. Сдача работы на проверку ее руководителю, доработка текста, графики.
10. Защита курсовой работы.

При выставлении оценки по курсовой работе учитывается: качество отчёта, знания студента по существу работы, оригинальность и творческий подход к реализации расчетов на компьютере (стиль программирования), а также своевременность выполнения всей работы и отдельных её этапов. Студент обязан не менее одного раза в

месяц информировать руководителя курсовой работы о выполненных этапах.

Отчёт по курсовой работе (пояснительная записка) должен содержать следующие разделы:

- титульный лист;
- задание по курсовой работе;
- аннотация;
- оглавление;
- введение;
- теоретические сведения;
- текст каждой задачи (постановка задачи);
- выбор метода решения и описание алгоритмов (блок-схемы);
- исходные данные;
- распечатки программ и результатов;
- результаты расчётов в виде графиков и таблиц;
- анализ решения задач, выводы;
- список использованной литературы.

На титульном листе указывается официальное название Университета, вид работы, наименование кафедры и название дисциплины, тема курсовой работы, фамилия и инициалы студента, шифр группы, дата оформления отчёта, должность, фамилия и инициалы руководителя работы, место для выставления оценки.

В аннотации следует привести краткие сведения о содержании работы (на русском и иностранном языках).

7. Рекомендательный библиографический список

1. Серогодский В.В., Прокди Р.Г. Excel 2010. Пошаговый самоучитель + справочник пользователя.- Наука и техника, 2012.

2. Штыков В.В. MathCAD. Руководство по решению задач для начинающих.- Либроком, 2013.

3. Васильев А.Н. Matlab. Самоучитель. Практический подход.- Наука и техника, 2012.

4. Зиборов В. Visual Basic 2012 на примерах.- БХВ-Петербург, 2013.

8. Варианты заданий для курсовой работы

Таблица 1.

Варианты для расчета углов треугольника

№	x_1	y_1	x_2	y_2	x_3	y_3
1	-2	2	4	2	8	6
2	-4	-1	5	2	9	9
3	-5	0	4	2	8	8
4	-3	-1	4	2	11	8
5	-2	-2	5	0	6	8
6.	-4	-2	1	1	9	7
7	-2	3	5	1	7	5
8	-1	0	3	4	8	6
9	-3	-1	5	0	8	5
10	-2	-1	2	0	7	7
11	-2	-2	4	0	7	8
12	-5	3	3	1	7	5
13	0	1	1	2	9	8
14	0	-1	4	0	10	6
15	-4	2	2	1	8	8
16	-4	1	5	2	10	5
17	-5	2	3	3	9	7
18	-3	-2	2	2	9	6
19	-1	0	2	1	10	8
20	-3	-1	4	4	8	6
21	-1	1	2	3	8	9
22	-5	0	6	4	7	6
23	-1	3	4	1	10	7
24	-2	2	4	0	7	7
25	-3	3	4	3	11	9
26	-3	1	6	3	8	7
27	-2	-1	4	2	8	7
28	-1	1	2	2	11	10
29	-4	3	1	3	11	8
30	-5	1	2	2	9	7

Таблица 2.

Варианты для вычисления неприступного расстояния

№	AC	δ		β		№	AC	δ		β	
		г	м	г	м			г	м	г	м
1	203,72	81	37	54	16	16	204,66	88	27	48	21
	236,13	82	6	49	9		165,49	88	23	54	31
	205,48	82	27	53	30		175,96	87	50	53	7
	227,02	80	36	51	14		240,53	82	46	46	13
2	240,15	86	49	50	48	17	182,66	88	51	49	42
	201,06	87	40	55	28		170,48	89	59	51	0
	240,06	81	47	53	43		220,50	80	1	48	24
	282,51	83	54	47	19		183,02	86	48	50	49
3	240,72	87	49	51	28	18	300,12	83	24	47	44
	271,57	84	27	49	42		242,69	86	26	52	41
	246,30	83	10	53	31		246,54	86	44	52	2
	253,03	88	12	49	47		262,66	83	20	52	3
4	215,13	87	23	48	12	19	212,33	81	15	48	16
	235,23	85	43	46	20		198,44	89	16	46	22
	231,15	84	19	47	34		168,02	85	23	53	30
	175,90	85	31	55	20		218,33	83	31	46	17
5	259,82	84	33	46	41	20	199,68	88	27	55	57
	209,00	84	27	53	34		308,54	83	19	45	42
	237,29	87	29	47	59		249,94	85	4	51	28
	238,66	84	55	49	10		268,03	86	32	48	31
6	194,75	81	16	53	30	21	164,39	87	5	53	7
	193,99	82	55	52	41		201,34	89	57	45	36
	185,94	86	23	51	59		197,82	86	20	47	59
	234,47	82	40	46	40		179,52	85	33	51	23
7	177,09	87	44	51	11	22	253,01	82	30	52	43
	162,75	86	43	54	17		246,48	83	45	52	50
	220,02	82	54	47	2		277,19	88	20	46	44
	192,98	85	2	50	7		305,61	82	1	46	47
8	258,42	88	22	49	59	23	191,78	86	44	52	5
	290,66	83	12	49	11		199,97	85	22	51	37
	280,83	87	57	47	46		185,75	86	26	53	13
	277,89	81	52	51	21		218,13	87	42	47	42

Продолжение таблицы 2

9	243,16	82	8	53	54	24	162,67	86	35	54	53
	257,29	84	48	50	35		198,59	81	32	51	39
	280,61	86	27	47	2		194,74	81	21	52	24
	300,29	81	51	47	7		241,10	81	51	45	5
10	230,03	85	5	50	7	25	173,38	87	35	55	5
	230,17	81	44	51	56		193,49	86	14	52	43
	253,61	85	14	46	60		175,85	88	18	54	13
	216,55	84	19	52	26		184,48	85	14	54	46
11	195,82	87	44	48	12	26	256,32	87	17	48	23
	214,51	83	22	47	37		240,64	88	28	49	36
	192,11	83	7	51	18		256,66	82	46	50	46
	205,62	85	54	47	41		215,16	88	52	52	35
12	277,43	84	37	46	46	27	188,37	83	7	54	4
	215,56	84	43	54	35		224,42	87	50	46	5
	209,99	86	48	54	4		231,48	87	6	45	31
	265,05	88	49	46	4		195,51	83	33	52	38
13	243,99	84	41	53	42	28	274,52	85	58	49	2
	253,88	88	34	50	13		267,08	86	39	49	30
	255,32	88	55	49	51		272,09	84	17	50	13
	241,93	81	41	55	48		260,59	89	42	48	32
14	226,81	82	54	48	20	29	282,01	83	17	48	18
	206,65	82	3	51	49		280,72	88	0	46	5
	207,38	87	46	48	35		246,22	87	26	50	17
	219,19	81	57	49	56		229,07	88	7	51	60
15	259,49	89	22	47	45	30	204,42	84	9	49	52
	276,77	82	37	49	20		193,56	82	31	52	31
	244,34	80	4	54	52		196,55	83	43	51	21
	267,27	85	10	49	7		230,74	81	25	47	18

Таблица 3.

Варианты для расчета превышения

№	l_1	S_{1-2}	ν		V_2
			г	м	
1	1,63	1836,40	0	-18,1	4
2	1,05	1855,70	0	-54,9	4
3	1,94	607,90	-2	-39,8	3
4	1,49	1496,60	-1	-58,1	2
5	1,36	2217,60	-1	-7,9	3
6	0,98	2485,60	0	-53,4	2
7	1,08	902,30	2	42,1	3
8	1,38	2053,90	0	1,1	4
9	1,91	2239,60	-1	-3,4	4
10	1,09	2178,60	1	47,7	3
11	1,56	573,50	0	-55,5	3
12	0,97	1080,50	1	41,6	3
13	1,20	1990,60	0	52,1	3
14	1,89	1098,90	-2	-26,5	2
15	1,19	703,60	-1	-51,7	2
16	1,09	2464,70	-2	-45,0	4
17	1,49	844,50	1	9,9	3
18	1,26	1889,20	1	5,5	3
19	1,46	640,50	-1	-42,6	4
20	1,34	1042,70	2	58,1	2
21	1,80	502,70	0	-14,0	4
22	1,13	2098,50	1	59,7	4
23	1,68	2365,00	0	-38,9	3
24	1,20	1272,30	0	-17,0	2
25	1,11	874,00	1	37,0	3
26	1,75	825,50	1	52,0	2
27	1,88	1867,30	-1	-18,5	2
28	1,60	2316,30	-2	-53,9	2
29	1,96	1412,70	1	51,1	4
30	1,76	641,80	0	56,5	3

Таблица 4.

Варианты для расчета координат точки по формулам Юнга

№	x_1	y_1	x_2	y_2	β_1			β_2		
					г	м	с	г	м	с
1	47	51	53	54	73	23	48,8	78	16	22,2
2	48	53	53	50	58	55	4,7	43	41	6,4
3	45	49	54	53	57	6	45,0	63	57	57,6
4	48	48	56	53	72	50	40,9	41	43	46,6
5	46	48	54	54	51	38	44,3	80	37	31,4
6	46	48	55	52	50	34	52,6	61	40	56,7
7	50	51	53	50	46	2	23,9	52	8	54,6
8	46	49	52	53	50	30	55,1	53	32	12,7
9	47	49	53	52	60	3	13,4	63	33	14,0
10	46	48	53	53	35	33	36,0	39	36	1,3
11	46	48	51	54	45	10	10,5	51	37	18,9
12	49	52	51	51	51	28	58,9	48	32	52,5
13	47	52	53	53	52	15	32,4	56	2	44,7
14	50	52	54	51	72	42	31,4	46	6	17,5
15	46	50	55	51	55	17	43,9	79	56	8,2
16	47	48	54	51	48	13	45,4	48	44	5,6
17	45	50	55	54	62	36	29,3	40	14	14,2
18	49	52	53	54	71	50	24,8	42	11	18,8
19	48	52	55	51	33	53	33,7	60	46	13,4
20	46	51	53	49	75	3	49,5	76	52	2,2
21	48	52	54	51	75	9	17,6	35	23	20,2
22	45	48	53	53	69	40	22,5	65	9	43,5
23	48	50	53	49	64	10	49,0	39	48	21,0
24	45	49	55	51	61	18	59,0	56	5	11,2
25	46	52	51	51	69	30	26,2	57	19	4,4
26	48	49	56	50	33	56	31,7	76	18	22,6
27	48	52	55	53	51	35	5,6	73	28	37,4
28	49	49	55	54	77	42	20,2	50	55	10,1
29	48	51	52	50	30	40	52,4	38	44	19,4
30	47	49	55	52	42	45	41,8	70	55	19,8

Таблица 5.

Варианты для обратной геодезической задачи

№	x_1	y_1	x_2	y_2
1	1,55	5,64	7,04	3,60
2	3,58	4,35	4,96	6,55
3	8,94	9,78	2,82	2,15
4	1,51	9,61	6,06	6,64
5	1,16	2,89	2,27	2,06
6.	9,90	5,97	7,36	2,45
7	6,85	7,04	1,18	2,64
8	9,07	1,19	5,43	3,60
9	1,81	6,18	8,82	3,65
10	2,93	2,87	2,12	3,52
11	6,73	6,24	6,16	6,28
12	8,19	6,49	5,99	9,24
13	7,81	8,09	1,48	7,95
14	8,07	9,44	6,45	7,28
15	9,72	4,85	6,47	1,28
16	1,75	8,40	7,14	7,57
17	2,59	7,49	4,54	1,31
18	8,60	1,24	6,82	1,98
19	3,15	9,31	9,54	3,47
20	7,92	3,71	8,74	7,86
21	6,31	8,86	3,09	4,31
22	5,16	1,74	7,08	9,27
23	5,18	9,42	5,61	6,94
24	9,62	5,02	4,39	7,27
25	2,04	1,71	5,21	6,78
26	9,92	9,51	3,93	7,11
27	4,79	5,99	5,84	4,56
28	3,41	4,01	2,19	9,85
29	4,05	5,86	1,69	5,76
30	8,76	6,57	4,93	5,90

Таблица 6.

Варианты для решения прямой угловой засечки по формулам Гаусса

№	x_1	y_1	x_2	y_2	α		β	
					г	м	г	м
1	49	52	51	50	60	39,0	307	17,0
2	47	48	52	51	25	8,0	345	24,0
3	48	50	52	50	22	58,0	333	53,0
4	49	52	51	54	41	13,0	323	51,0
5	48	50	56	49	45	0,0	332	18,0
6	49	51	52	50	20	27,0	333	25,0
7	46	52	55	53	42	15,0	336	26,0
8	46	51	54	50	31	56,0	341	17,0
9	45	49	53	53	34	38,0	347	57,0
10	48	48	52	52	42	53,0	335	31,0
11	48	51	54	51	50	1,0	333	23,0
12	49	53	55	51	36	41,0	312	16,0
13	50	49	53	50	52	12,0	300	45,0
14	45	48	56	52	45	9,0	332	37,0
15	50	52	56	52	17	24,0	332	0,0
16	48	50	52	51	34	58,0	309	27,0
17	47	50	56	53	14	3,0	308	12,0
18	49	53	54	52	18	45,0	318	27,0
19	49	53	51	52	37	55,0	331	9,0
20	48	48	52	52	21	26,0	325	12,0
21	50	50	52	54	54	16,0	349	12,0
22	45	53	54	53	59	7,0	309	27,0
23	46	50	52	51	16	3,0	304	1,0
24	49	49	53	51	36	0,0	347	15,0
25	47	51	52	53	22	14,0	326	52,0
26	47	53	52	53	36	47,0	304	1,0
27	48	50	53	53	22	47,0	329	54,0
28	46	49	51	51	40	15,0	311	2,0
29	47	52	54	51	20	48,0	319	19,0
30	45	52	54	49	34	39,0	341	21,0

Таблица 7.

Варианты для решения обратной угловой засечки по формулам Пранис-Праневича

№	x_1	y_1	x_2	y_2	x_3	y_3	α		β	
							г	м	г	м
1	3.33	1.43	1.28	4.39	1.54	7.35	20	21.3	19	57.9
2	2.79	0.77	0.55	3.65	1.79	7.14	20	57.3	18	59.4
3	2.82	0.75	1.29	3.77	2.09	7.29	19	54.2	18	19.9
4	2.66	1.07	1.16	3.97	1.56	7.12	20	19.8	18	30.3
5	2.91	1.00	1.03	4.16	2.46	7.05	20	56.4	18	9.7
6	3.24	1.28	1.15	4.07	2.01	7.28	20	19.5	18	55.5
7	3.49	0.68	1.10	3.61	2.04	6.68	20	41.1	18	44.2
8	3.21	1.32	1.29	4.03	1.61	7.35	19	5.1	19	32.9
9	2.85	0.90	1.06	4.50	1.66	6.87	20	10.9	19	16.6
10	2.98	0.97	0.70	4.30	2.10	6.98	19	50.8	18	16.9
11	2.72	1.21	1.14	3.52	2.34	6.67	19	6.5	18	17.7
12	3.25	0.77	1.35	4.30	2.36	7.32	20	21.3	19	56.1
13	2.64	1.23	0.57	4.05	1.57	7.45	19	56.0	19	23.9
14	2.82	1.38	1.13	4.19	2.00	7.29	20	5.8	18	11.3
15	3.04	1.25	0.57	3.94	1.65	7.00	20	17.6	19	25.4
16	2.51	1.46	0.82	3.98	2.04	7.24	20	3.7	18	26.5
17	3.32	1.33	1.09	4.20	1.72	7.41	19	22.6	18	47.4
18	2.72	0.83	0.98	4.20	1.81	6.99	20	51.2	19	5.74
19	2.86	1.41	1.00	3.86	1.80	6.94	20	17.7	18	38.3
20	2.60	1.12	0.57	4.04	1.93	7.37	20	3.9	19	49.3
21	2.88	0.89	0.70	4.07	1.59	6.71	20	36.3	19	49.1
22	3.41	0.84	1.19	4.01	2.43	6.67	19	16.2	18	20.8
23	3.33	0.67	1.40	4.10	2.14	7.49	19	45.1	19	44.6
24	2.82	1.44	1.08	4.48	1.66	6.93	20	32.6	18	28.2
25	2.93	1.15	1.11	4.05	2.47	6.81	20	32.3	18	3.1
26	3.48	0.94	1.42	4.41	1.96	6.83	19	37.2	18	26.7
27	3.34	0.58	0.57	3.86	2.10	7.02	19	42.6	19	24.3
28	2.61	0.58	1.09	3.64	1.88	7.11	20	55.1	19	30.5
29	2.89	1.47	1.23	4.28	2.15	7.34	20	13.7	19	15.8
30	2.94	0.92	1.48	4.38	1.68	7.09	19	30.7	18	58.2