|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | | |
|  | | | |
| Кафедра прикладной математики | | | |
|  | | | |
| Практическое задание № 2 | | | |
| по дисциплине «Программирование вычислений» | | | |
| **Вычисление заданного набора геометрических характеристик треугольника** | | | |
|  | | | |
|  | Группа | ПМ-81 |
| Бригада | 4 |
| Студенты | Басов Денис |
|  | Юрганов Егор |
|  |  |
| Преподаватель | Рояк Михаил Эммануилович |
|  |  |
|  | | | |
| Новосибирск | | | |

1. **Цель работы:**

Изучение операторов ввода-вывода и форматирования текста на языке FORTRAN. Знакомство с такими понятиями вычислительной математики, как «конечная арифметика», «значащие цифры», «верные значащие цифры», «абсолютная и относительная погрешности», «накопление погрешности вычислений», «округление».

1. **Задание:**

Написать программу, реализующую на языке ФОРТРАН построение таблицы значений заданной функции двух переменных

Входные данные: значения 𝑥𝑚𝑖𝑛, 𝑥𝑚𝑎𝑥, ∆𝑥, 𝑦𝑚𝑖𝑛, 𝑦𝑚𝑎𝑥, ∆𝑦

Выходные данные: форматированная таблица значений функции

1. **Математическая модель:**

1) Заданная функция:

2) Область определения функции:

1. **Текст программы:**

subroutine input()

common /par/xmin,xmax,ymin,ymax,sx,sy

open(1,FILE='in.txt')

read(1,\*,err=1,end=1) xmin,xmax,sx,ymin,ymax,sy

print \*,'xmin=',xmin,'xmax=',xmax,'sx=',sx

print \*,'ymin=',ymin,'ymax=',ymax,'sy=',sy

if((xmin.gt.xmax.or.ymin.gt.ymax).or.

\*(sx.le.0.0.or.sy.le.0.0)) goto 2

goto 3

1 continue

pause 'Reading error!!!'

stop

2 continue

pause 'Incorrect values!!!'

stop

3 continue

close(1)

end

real function func(fx,fy)

common /par/xmin,xmax,ymin,ymax,sx,sy

pi=3.1415927

x=fx

y=fy

x=(x\*pi)/180

y=(y\*pi)/180

func=tan(x)/cos(y)

end

subroutine invstep(a,b)

common/var/iflag,iflag2,inv,max

Character \*13 stra,strb

write (stra,'(e10.4)') a

write (strb,'(e10.4)') b

inv = 0

max = 0

if (stra.ne.strb) inv=1

if ((stra.ne.strb).or.(stra.lt.strb)) max=1

end

subroutine proverka(zx,zy)

common/par/xmin,xmax,ymin,ymax,sx,sy

common/var/iflag,iflag2,inv,max

iflag=0

iflag2=0

temp1=2

temp2=0

call invstep(zy,90.0)

if(inv) then

call invstep(zy,270.0)

if(inv.ne.1) then

iflag=1

temp1=1

endif

else

iflag=1

temp1=1

endif

call invstep(zx,90.0)

if(inv) then

call invstep(zx,270.0)

if(inv.ne.1) iflag=1

else

iflag=1

endif

call invstep(zx,0.0)

if(inv) then

call invstep(zx,180.0)

if(inv.ne.1) temp2=1

else

temp2=1

endif

if(temp1.eq.temp2) iflag2=1

end

subroutine stroka(y)

common/par/xmin,xmax,ymin,ymax,sx,sy

common/var/iflag,iflag2,inv,max

common/str/n,k

220 format(/\)

20 format(E11.4,'|'\)

21 format(E11.4,'|')

23 format(12('-'\))

25 format(12('-'))

24 format(' inf |'\)

26 format(' inf |')

33 format(' nan |'\)

i=0

x=xmin

do while(x.le.xmax)

if(abs(x).lt.abs(sx\*1E-04)) x=0

if(i.EQ.0) then

call proverka(x,y)

if(iflag2) then

write(2,33)

else

if(iflag) then

write(2,24)

else

write(2,20) func(x,y)

endif

endif

else

call invstep(x,xmin+(i-1)\*sx)

if(inv) then

call proverka(x,y)

if(iflag2) then

write(2,33)

else

if(iflag) then

write(2,24)

else

write(2,20) func(x,y)

endif

endif

endif

endif

i=i+1

x=xmin+i\*sx

enddo

call invstep(x,xmax)

if(max) then

write(2,220)

else

if(iflag) then

write(2,26)

else

write(2,21) func(xmax,y)

endif

endif

do j=0,n,1 !line separator

write(2,23)

enddo

write(2,25)

end

program main

common/par/xmin,xmax,ymin,ymax,sx,sy

common/var/iflag,iflag2,inv,max

common/str/n,k

220 format(/\)

20 format(E11.4,'|'\)

21 format(E11.4,'|')

22 format(A3,8x,'|'\)

23 format(12('-'\))

25 format(12('-'))

call input

open(2,FILE='out.txt')

n=0

k=0

i=0

zx = xmin !sum step x

do while(zx.lt.xmax)

if(abs(zx).lt.sx\*1E-04) zx=0

if(i.EQ.0) then

n=n+1

else

call invstep(zx,xmin+(i-1)\*sx)

if(inv) n=n+1

endif

i=i+1

zx = xmin+i\*sx

enddo

write(2,22) 'Y\X' !title

i=0

x=xmin

do while(x.le.xmax)

if(abs(x).lt.abs(sx\*1E-04)) x=0

if(i.EQ.0) then

write(2,20) x

else

call invstep(x,xmin+(i-1)\*sx)

if(inv) write(2,20) x

endif

i=i+1

x=xmin+i\*sx

enddo

call invstep(x,xmax)

if(max) then

write(2,220)

else

write(2,21) xmax

endif

do i=0,n,1

write(2,23) !line separator

enddo

write(2,25)

i=0

y=ymin

do while(y.le.ymax) !filling

if(abs(y).lt.abs(sy\*1E-04)) y=0

if(i.EQ.0) then

write(2,20) y

call stroka(y)

else

call invstep(y,ymin+(i-1)\*sy)

if(inv) then

write(2,20) y

call stroka(y)

endif

endif

i=i+1

y=ymin+i\*sy

enddo

call invstep(y,ymax)

if(max) then

write(2,220)

else

write(2,20) ymax

call stroka(ymax)

endif

close(2)

print \*,'Finished!!!'

pause

end

1. **Система тестов**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | xmin | xmax | xstep | ymin | ymax | ystep | Особенность |
| 1 | 0 | 10 | 1 | 45 | 50 | 1 | Тангенс 0 |
| 2 | 10 | 20 | 2 | 90 | 180 | 10 | Деление на 0 |
| 3 | 45 | 60 | -1 | 0 | 1000 | 1 | Шаг<0 |
| 4 | 45 | 60 | 1 | 0 | 1000 | -11 | Шаг не попадает в Max |
| 5 | 0 | 0 | 1 | 0 | 1000 | 11 | Min=Max |
| 6 | 45 | 60 | 1 | 10 | 0 | 11 | Min>Max |

1. **Протокол исследований**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | xmin | xmax | xstep | ymin | ymax | ystep | Особенность |
| 1 | 45 | 47 | 1 | 89 | 90.01 | 0.01 | f(x,y)=∞ |
| 2 | 89 | 90.01 | 0.01 | 0 | 2 | 1 | f(x,y)=∞ |
| 3 | 45 | 47 | 1 | 89 | 90.01 | 0.001 | Невидимый шаг |
| 4 | -1.009e-7 | 1.009e-7 | 1e-11 | 0 | 2 | 1 | Невидимый шаг |
| 5 | 89 | 90.01 | 0.001 | 0 | 2 | 1 | Невидимый шаг |
| 6 | -5e-7 | 1e-7 | 1e-7 | 89.95 | 90.01 | 0.01 | 0/0 |
| 7 | -99e-7 | 1e-7 | 1e-7 | 0 | 2 | 1 | 0 аргумента x |
| 8 | 45 | 49 | 1 | -99e-7 | 1e-7 | 1e-7 | 0 аргумента y |