МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»

(БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Дисциплина: Вычислительная математика

Лабораторная работа №4

Решение системы двух нелинейных уравнений с двумя неизвестными методом Ньютона

|  |  |
| --- | --- |
|  | Выполнила: ст. группы ПВ-21  Зановская А.И.  Проверила: Бондаренко Т.В. |

Белгород

2017

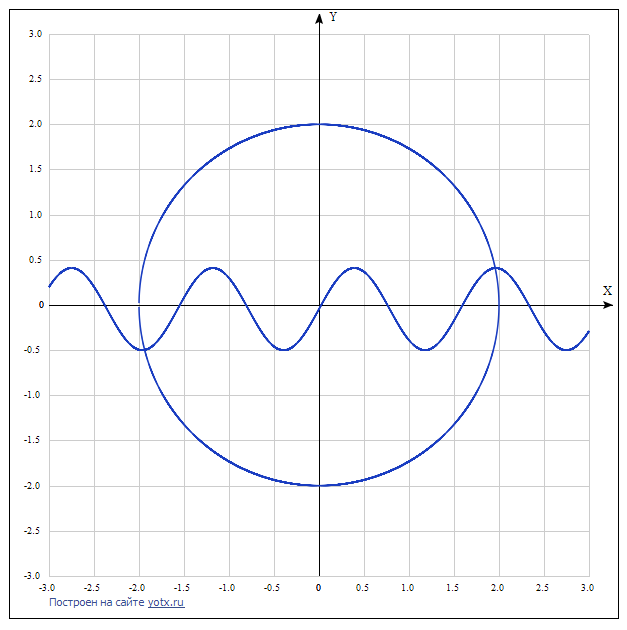
**Цель работы**: изучение и получение практических навыков приближенного решения систем двух нелинейных уравнений с двумя неизвестными методом Ньютона.

**Вариант 10**

**Задания к работе**

1. Записать для уравнений системы соответствующего варианта задания функции F(x, y) и Φ(x, y).

Построить в одной системе координат графики функций F(x, y) и Φ(x, y).

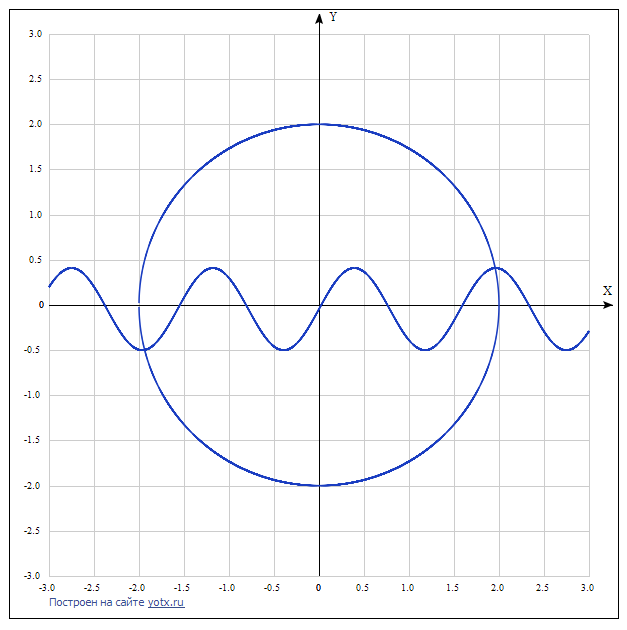


Найти точки пересечения графиков функций F(x, y) и Φ(x, y), которые соответствуют решениям системы уравнений.

(1,96;0,41)

(-1,94;-0,497)

2. Определить область содержащую одну из точек пересечения графиков функций F(x, y) и Φ(x, y). Выбрать начальное приближение решения системы уравнений ― точку М0 с координатами (x0; y0), принадлежащую выбранной области.



3. Найти частные производные первого порядка по переменным x, y для функций F(x, y) и Φ(x, y) и вычислить значения производных в точке М0. Записать линеаризованную систему, соответствующую исходной нелинейной системе, для выбранного начального приближения М0(x0; y0).

4. Выполнить один шаг численного метода решения системы двух нелинейных уравнений с двумя неизвестными «вручную» и найти следующее приближение к решению системы уравнений (x1; y1). Выполнить проверку правила остановки с точностью ε=0,001.

5. Описать в модуле логическую функцию для приближенного решения системы двух нелинейных уравнений с двумя неизвестными методом Ньютона с заданной точностью ε. Входными данными являются: - функции уравнений системы и их частные производные (f (x, y), ∂f / ∂x, ∂f / ∂y),(g(x, y), ∂g / ∂x, g / ∂y) ; - начальное приближение (x0; y0); - точность решения ε; - максимальное число итераций n. Функция возвращает значение «истина», если приближенное решение системы с заданной точностью получено за число итераций, не превышающее n, при этом вычисленная пара (x; y) – приближенное решение системы двух уравнений с точность решения ε. В противном случае функция возвращает значение «ложь».

#ifndef BOOL\_CH\_H\_INCLUDED

#define BOOL\_CH\_H\_INCLUDED

typedef float (\*func)(float, float);

int ch (func F,func dF\_dx,func dF\_dy,func Fi,func dFi\_dx,func dFi\_dy, int n, float e, float x, float y)

{

int i;

float l,h,s1,s2;

float \*\*pa;

pa=(float \*\*) calloc(2,sizeof(float \*));

for (i=0;i<2;i++)

pa[i]=(float \*) calloc(3,sizeof(float));

i = 0;

while (i<n && s1+s2>e)

{

pa[0][0]=dF\_dx(x,y);

pa[0][1]=dF\_dy (x,y);

pa[0][2]=-F(x,y);

pa[1][0]=dFi\_dx(x,y);

pa[1][1]=dFi\_dx(x,y);

pa[1][2]=-Fi(x,y);

decision (pa,2,3);

back\_stroke (pa,2,3);

h=pa[0][2];

l=pa[1][2];

x+=h;

y+=l;

s1=fabs(F(x,y));

s2=fabs(Fi(x,y));

}

if(s1+s2<e)

{

printf ("\nЗаданная точность достигнута");

printf ("\nx = %f, y = %f",x,y);

return 1;

}

else

{

printf ("\nЗаданная точность недостигнута");

printf ("\nx = %f, y = %f",x,y);

return 0;

}

}

float F (float x, float y)

{

return (sin(4\*x)-2.2\*y-0.1);

}

float dF\_dx (float x, float y)

{

return (4\*cos(4\*x));

}

float dF\_dy (float x, float y)

{

return (-2.2);

}

float Fi (float x, float y)

{

return (x\*x+y\*y-4);

}

float dFi\_dx (float x, float y)

{

return (2\*x);

}

float dFi\_dy (float x, float y)

{

return (2\*y);

}

void subs (float \*\*a, int m, int n,int i)

{

int j,y;

float k;

y=i;

for (++i;i<m;i++)

{

k=-a[i][y]/a[y][y];

for (j=0;j<n;j++)

a[i][j]+=k\*a[y][j];

}

}

void sub (float \*\*a, int m, int n,int i)

{

int j,y;

float k;

y=i;

for (--i;i>=0;i--)

{

k=-a[i][y]/a[y][y];

for (j=0;j<n;j++)

a[i][j]+=k\*a[y][j];

}

for (j=y+1;j<n;j++)

a[y][j]/=a[y][y];

a[y][y]=1;

}

void swap (float \*\*a, int x, int y, int n)

{

int i; float \*k;

k=a[x];

a[x]=a[y];

a[y]=k;

}

//m - кол-во строк, j - проверяемый столбец

int search (float \*\*a, int m, int j)

{

int i, nom;

float max;

nom = j; max = a[j][j];

for (i=j+1;i<m;i++)

if (fabs(a[i][j])-fabs(max)>0.001)

{

max=fabs(a[i][j]);

nom=i;

}

return nom;

}

int decision (float \*\*a, int m, int n)

{

int i,j=0,nom,t=0;

for (i=0;i<m-1;i++)

{

if (a[i][i]==0)

return 0;

nom=search(a,m,j);

if (nom!=i)

{

swap(a,i,nom,n);

}

subs(a,m,n,i);

j++;

}

}

void back\_stroke (float \*\*a, int m, int n)

{

int i,j,g,k;

float sum;

for (k=m;k<n;k++)

{

for (i=m-1;i>=0;i--)

{

sum=0;

for(j=m-1;j>i;j--)

sum+=a[i][j]\*a[j][k];

a[i][k]=(a[i][k]-sum)/a[i][i];

}

}

}

#endif // BOOL\_CH\_H\_INCLUDED

6. Составить программу для решения системы двух нелинейных уравнений соответствующего варианта задания.

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

#include <math.h>

#include "bool\_ch.h”

int main()

{

setlocale (LC\_ALL, "Rus");

float x,y,e;

int n;

printf ("x = "); scanf ("%f",&x);

printf ("y = "); scanf ("%f",&y);

printf ("e = "); scanf ("%f",&e);

printf ("Максимальное количество итераций: "); scanf ("%d",&n);

return (ch(F,dF\_dx,dF\_dy,Fi, dFi\_dx, dFi\_dy, n, e, x, y));

}