**МИНОБРНАУКИ РОССИИ**

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 №6

тема: «Одномерная минимизация функции»

|  |  |
| --- | --- |
|  | Выполнил: ст. группы ПВ-21  Зановская Анна Ивановна  Проверил: Бондаренко Т.В. |

Белгород

2017

**Цель работы:** изучить методы нахождения приближенного решения задачи одномерной

минимизации функции одной переменной, и получить практические навыки их применения.

**Задания к работе:**

1. Найти область определения заданной функции у = f(x) и построить её график, используя

равномерную сетку значений хi (шаг сетки выбрать самостоятельно).

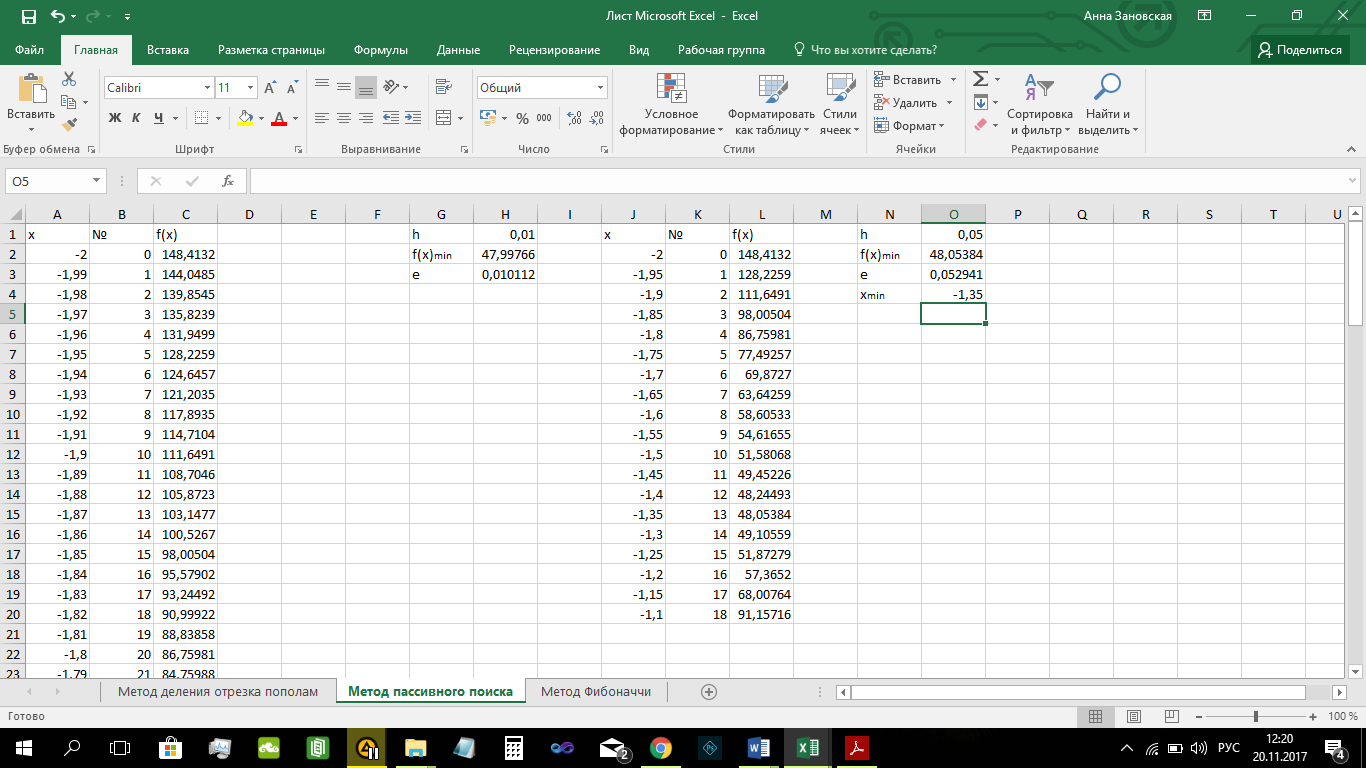
2. Найти промежутки унимодальности функции у = f(x), используя построенный график.

3. Найти первую y´=f´(x) и вторую y´´= f´´ (x) производные заданной функции у = f(x).

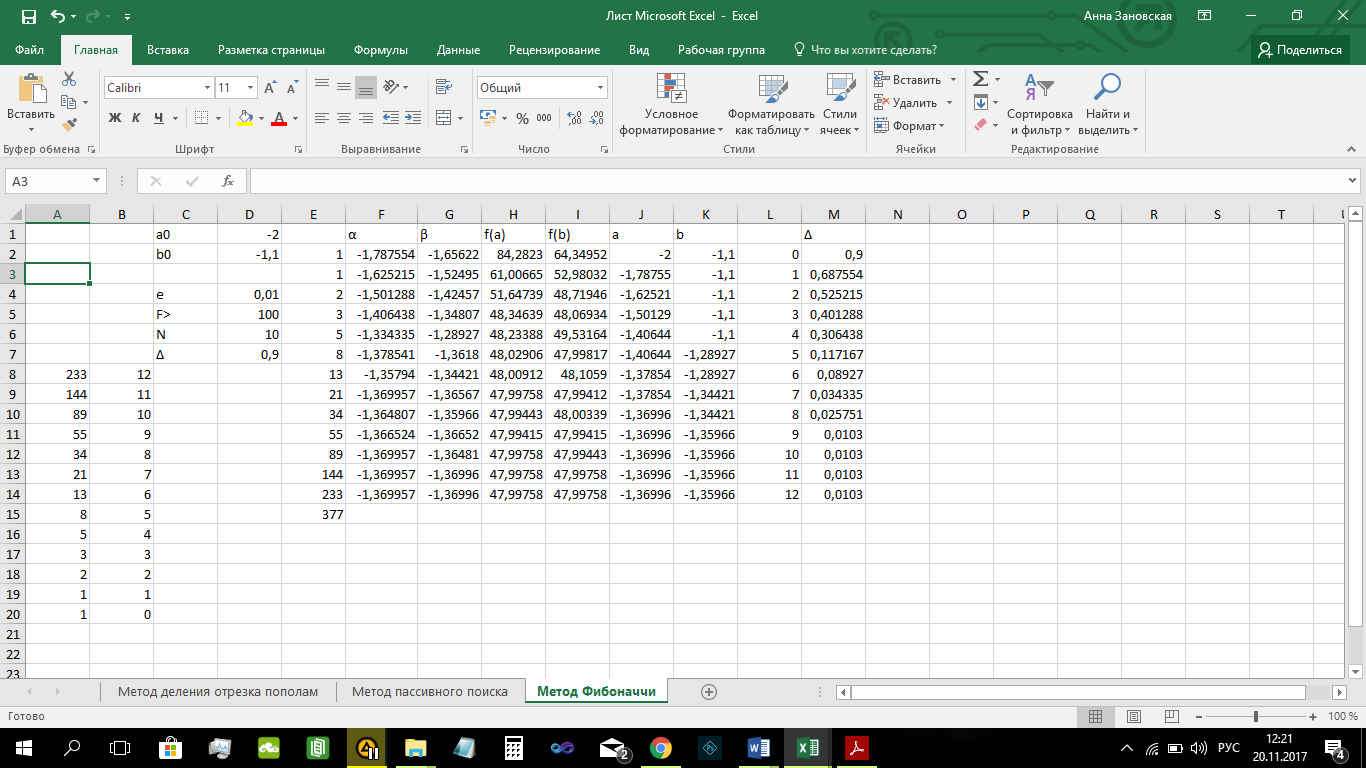
4. Найти точное решение задачи одномерной минимизации ― минимум функции у = f(x), точку хТ, и минимальное значение функции 𝑚𝑖𝑛(𝑓(𝑥Т)).

5. Найти приближенное решение задачи одномерной минимизации, точку х̃ такую, что 𝑥Т ≈ х̃ вручную, используя численные методы одномерной минимизации:

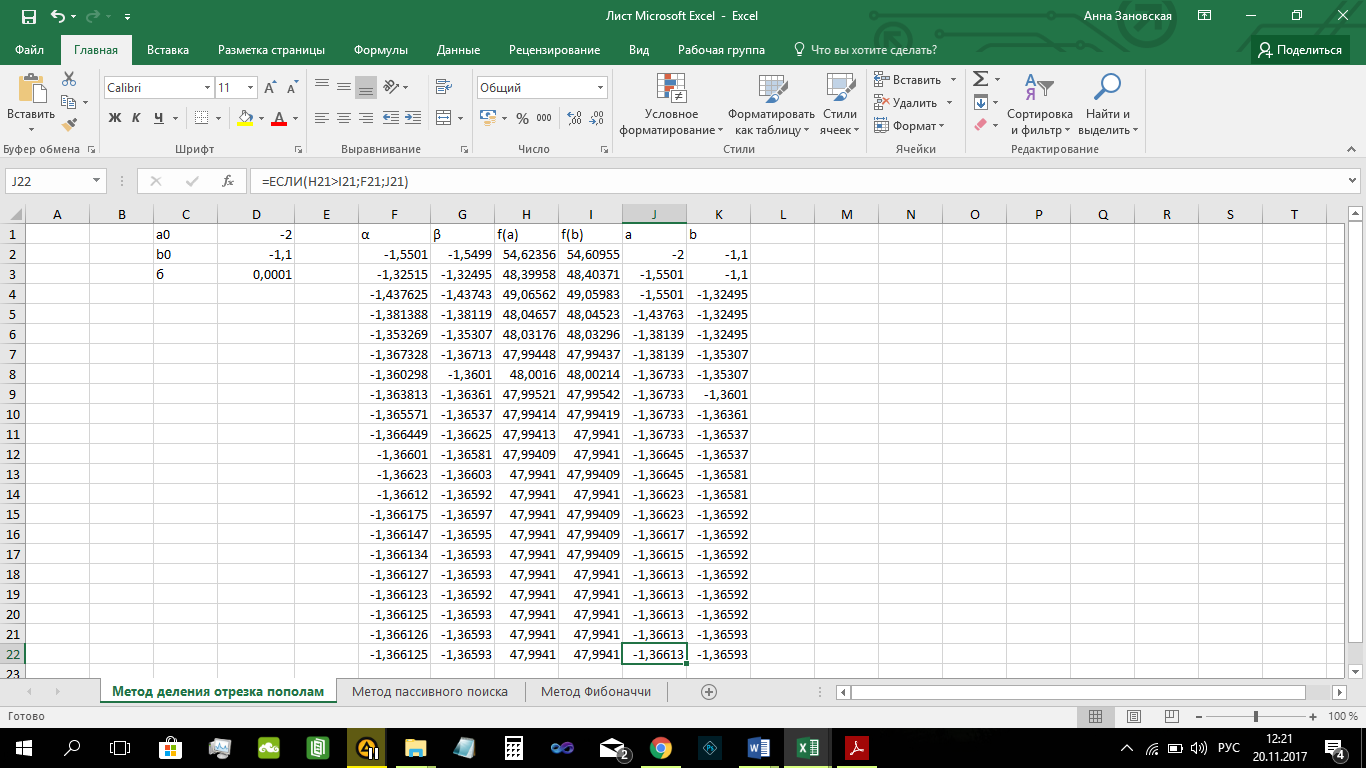
* метод оптимального поиска;



* метод, основанный на использовании чисел Фибоначчи;



* метод деления отрезка пополам;



с точностью ε =0,01.

6. Определить абсолютную Δ и относительную δ погрешность решения задачи одномерной минимизации для каждого из используемых численных методов. Представить полученные результаты в виде таблицы (см. табл. 2).

|  |  |  |  |
| --- | --- | --- | --- |
|  | Задача одномерной минимизации | | |
|  | метод оптимального  поиска | метод чисел  Фибоначчи | метод деления  отрезка пополам |
| Δ | 0,004 | 0,004 | 0,0001 |
| δ | 0,003% | 0,03% | 0,007% |

7. Описать в модуле функции, которые возвращают приближенные значения минимума

функции у = f(x) для заданного промежутка унимодальности X ⸦ R с заданной точностью ε каждым из рассмотренных численных методов: метод оптимального поиска; метод, основанный на использовании чисел Фибоначчи; метод деления отрезка пополам.

float f (float x)

{

return -((float)exp(x\*x+1)/(x+1));

}

float PasSearch (float a, float b, float e)

{

int n,i,min;

n=(b-a)/e;

float \*\*p;

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

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

p[i]=(float \*)calloc(n,sizeof(float));

//первая строка - х, вторая - у

p[0][0]=a;

p[1][0]=f(a);

min=0;

for (i=1;i<n;i++)

{

p[0][i]=p[0][i-1]+e;

p[1][i]=f(p[0][i]);

if(p[1][min]>p[1][i])

min=i;

}

return p[0][min];

}

float DivTo2 (float a, float b, float e)

{

float Al=0,Be=0;

int i=0,t=100;

while (fabs(b-a)>e && i<t)

{

Al=(float)((a+b)/2-e);

Be=(float)((a+b)/2+e);

if (f(Al)<=f(Be))

b=Be;

else

a=Al;

i++;

}

return (Al);

}

float Fib (float a, float b, float e)

{

int k,k1,k2,t=0,n=0,i,j;

float Al,Be,d;

k=(int)(1/e);

k1=k2=1; n=2;

while (t<k)

{

t=k1+k2;

k1=k2;

k2=t;

n++;

}

n+=1;

int \*F;

F=(int \*)calloc (n,sizeof (int));

F[0]=F[1]=1;

i=2;

while (i<n)

{

F[i]=F[i-1]+F[i-2];

i++;

}

n-=2;

k=0;

d=(float)(b-a);

while (d>e)

{

Al=(a+(float)F[n-k-1]/F[n-k+1]\*d);

Be=(a+(float)F[n-k]/F[n-k+1]\*d);

if (f(Al)<=f(Be))

b=Be;

else

a=Al;

k++;

d=(float)(b-a);

}

return (a);

}

8. Составить программу для вычисления приближенного решения задачи одномерной

минимизации для заданного варианта задания с использованием функций, описанных в модуле.

int main()

{

setlocale(LC\_ALL,"Rus");

float a,b,e=0.001, t,x;

printf ("Введите а ");

scanf ("%f", &a);

printf ("Введите b ");

scanf ("%f", &b);

printf ("Введите точное значение ");

scanf ("%f", &t);

//t=f(t);

x=PasSearch(a,b,e); //x=f(x);

printf ("\nMeтод пассивного поиска");

printf ("\nх = %f D = %f d = %f%% ",x, (float)fabs(x-t),(float)fabs((x-t)/t\*100));

printf ("\nMeтод деления отрезка пополам");

x=DivTo2(a,b,e);

printf ("\nх = %f D = %f d = %f%% ",x, (float)fabs(x-t),(float)fabs((x-t)/t\*100));

printf ("\nMeтод чисел Фибоначчи");

x=Fib(a,b,e);

printf ("\nх = %f D = %f d = %f%% ", x,(float)fabs(x-t),(float)fabs((x-t)/t\*100));

}

