МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждениевысшего образования
«Московский государственный технологический университет «СТАНКИН»(ФГБОУ ВО «МГТУ «СТАНКИН»)

Институт информационных технологий

Кафедра информационных технологий и вычислительных систем

ОТЧЕТ О ВЫПОЛНЕНИИ

ИНДИВИДУАЛЬНОГО ЗАДАНИЯ ПО ДИСЦИПЛИНЕ

«Методы оптимизации»

СТУДЕНТА *2* КУРСА *бакалавриата* ГРУППЫ *ИДБ-22-04*

(уровень профессионального образования)

Макаров Андрей Олегович

НА ТЕМУ

«Численные методы многомерной оптимизации 2 порядка»

Направление: 09.03.01 Информатика и вычислительная техника

Профиль подготовки: «Разработка программных комплексов в рамках цифровой трансформации деятельности предприятий»

Отчет сдан « ______ » _______________20 ___г.

Оценка ______________________________

Преподаватель Палванов М.Р. ____________

(Ф.И.О., должность, степень, звание.) *(подпись)*

МОСКВА 2024

Метод второго порядка. Метод Ньютона

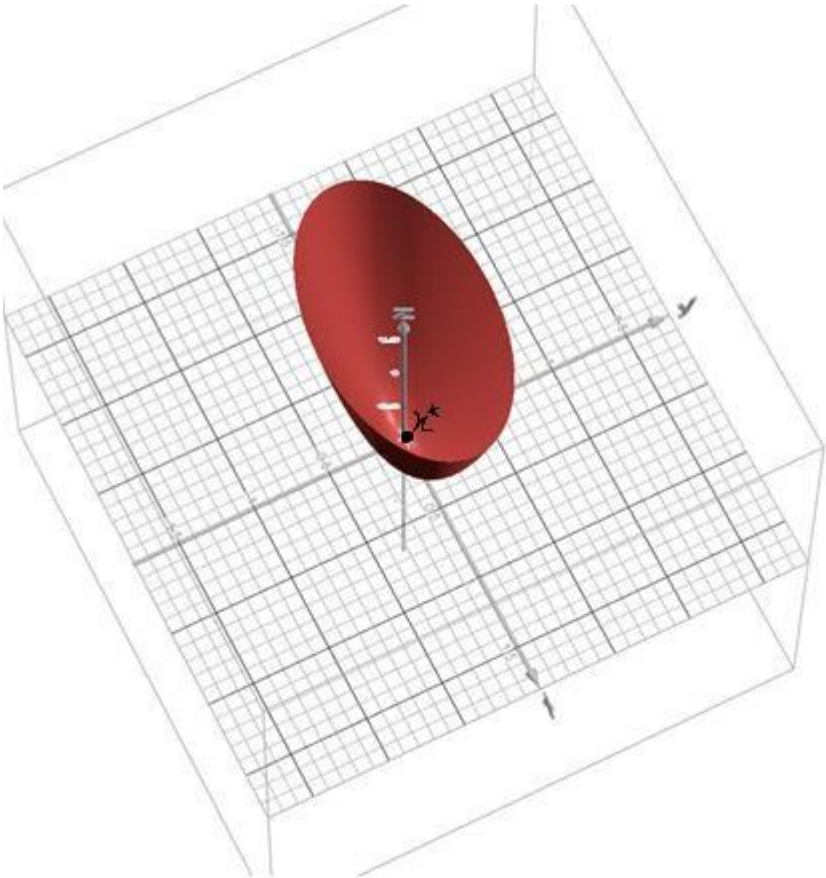
Дано:

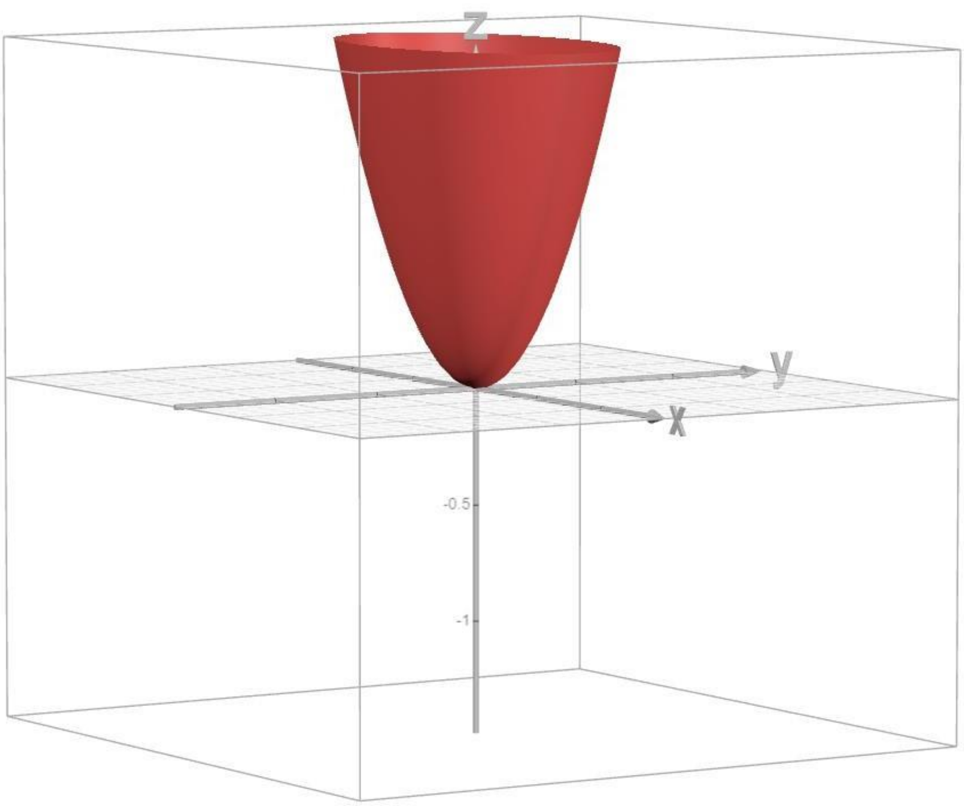
$$f(x) = 2x^2 + 0.2xy + 2y^2$$

$x^0 = (1.5; 0.5)$ ε = 0,15 ε = 0,2 M =10

Минимум функции был сразу найден с помощью программного кода,

График функции из индивидуального задания:





Ниже предложен результат работы программы, написанной на C++.

```c
#include <stdio.h>
#include<math.h>


int main(void){
    double x1,x2,x1x2;
    printf("Введите коэфиценты перед x1^2 x1x2 x2^2\n");
    scanf("%lf%lf%lf",&x1,&x1x2,&x2);
    printf("Уровнение =   %.1lf*x^2 %.1lf*x1x2 %.1lf*x2^2\n",x1,x1x2,x2);
    double e1,e2,m,x0,x01;
    printf("Введите x0 e1 e2 M\n");
    scanf("%lf %lf %lf %lf %lf",&x0,&x01,&e1,&e2,&m);


    double dx1,dx2;
    dx1=2*x1;
    dx2=2*x2;
    int k=0;
    double t=0.5;
    double x11,x12,x21,x22,x00,x001=0,x5,x6;
    double hess[2][2];
    double dk[2];
    x00=x0; //k
    x001=x01;
    x5=x0;   //k-1
    x6=x01;


    m2:
    x11=dx1*x00+x1x2*x001;  //f(x)k
    x12=dx2*x001+x1x2*x00;
    printf("k = %d  ------  f(x)=[%f;%f]   xk= [%f;%f]   -----------------\n",k,x11,x12,x00,x001);


    if((sqrt(x11*x11+x12*x12)<e1)){
        x0=x00;
        x01=x001;
        printf("Градиент xk < e1 \n");
    }
    else {
        printf("Градиент xk > e1 \n");
        printf("Вычислим ||f(x)|| > e =   %f > %f\n",sqrt(x11*x11+x12*x12),e1);
```


        printf("Вычислим ||f(x)|| > e =   %f > %f\n",sqrt(x11*x11+x12*x12),e1);

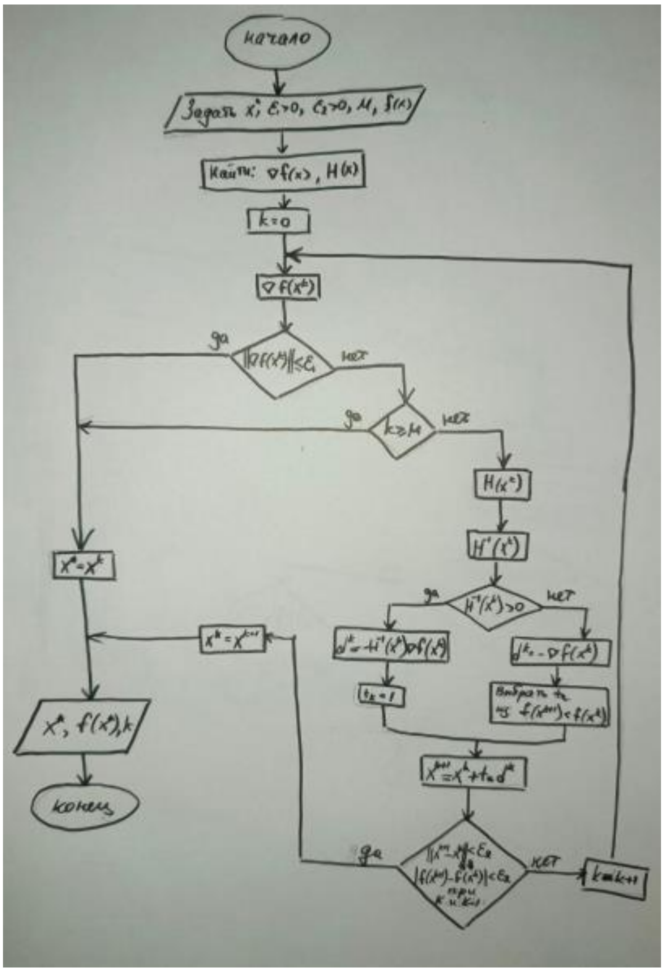

```c
        if(k>=m)
        {
            x0=x00;
            x01=x001;

        }
        else
        {

            hess[0][0]=dx1;
            hess[0][1]=x1x2;
            hess[1][0]=x1x2;
            hess[1][1]=dx2;

            double opr =hess[0][0]* hess[1][1]-hess[0][1]*hess[1][0];
            double trans[2][2];
            trans[0][0]=hess[1][1];
            trans[0][1]=-hess[1][0];
            trans[1][0]=-hess[0][1];
            trans[1][1]=hess[0][0];


            hess[0][0]=trans[0][0]*(1/fabs(opr));
            hess[0][1]=trans[1][0]*(1/fabs(opr));
            hess[1][0]=trans[0][1]*(1/fabs(opr));
            hess[1][1]=trans[1][1]*(1/fabs(opr));
            printf("hess[0][0] = %f hess[0][1]= %f hess[1][0]= %f hess[1][1] = %f\n",hess[0][0],hess[0][1],hess[1][0],hess[1][1]);
            if((hess[1][1]>0)&&((hess[0][0]* hess[1][1]-hess[0][1]*hess[1][0])>0)){
                dk[0]=-1*(hess[0][0]*x11+hess[0][1]*x11);
                dk[1]=-1*(hess[1][0]*x12+hess[1][1]*x12);
                t=1;
                x21=x00+dk[0];
                x22=x001+dk[1];

            }
            else
            {
                dk[0]=-1*x11;
                dk[1]=-1*x12;
                m1:
```

```c
int main(void){
            m1:
            x21=x00+t*dk[0];
            x22=x001+t*dk[1];
            if((x1*x21*x21+x1x2*x21*x22+x2*x22*x22)-(x1*x00*x00+x1x2*x00*x001+x2*x001*x001)<0);
                else
                {
                    t=t/2;
                    goto m1;
                }

        }

        if(  (sqrt((x21-x00)*(x21-x00)+(x22-x001)*(x22-x001))<e2) &&
            (abs(((x1*x21*x21+x1x2*x21*x22+x2*x22*x22)-(x1*x00*x00+x1x2*x00*x001+x2*x001*x001)))<e2)&&(sqrt((x00-x5)*(x00-x5)+
            (x001-x6)*(x001-x6))<e2) &&  (abs(((x1*x00*x00+x1x2*x00*x001+x2*x001*x001)-(x1*x5*x5+x1x2*x5*x6+x2*x6*x6)))<e2)){
            printf("Проверим на выполение обоих условий для k и k=k-1 ||xk+1 - xk ||= %f > %f  ------   |f(xk+1) -  f(xk) |=%f > %.5fl
                \n",sqrt((x21-x00)*(x21-x00)+(x22-x001)*(x22-x001)),e2,abs(((x1*x21*x21+x1x2*x21*x22+x2*x22*x22)-
                (x1*x00*x00+x1x2*x00*x001+x2*x001*x001))),e2);
            printf("Условие выполнено \n");
            x0=x21;
            x01=x22;
        }
            else{
                printf("Проверим на выполение обоих условий ||xk+1 - xk ||= %f > %f  ------   |f(xk+1) -  f(xk) |=%f > %f
                    \n",sqrt((x21-x00)*(x21-x00)+(x22-x001)*(x22-x001)),e2,abs(((x1*x21*x21+x1x2*x21*x22+x2*x22*x22)-
                    (x1*x00*x00+x1x2*x00*x001+x2*x001*x001))),e2);
                printf("Условие не выполнено \n");
                k=k+1;
                x00=x21;
                x001=x22;
                x5=x00;
                x6=x001;
                printf("\n");
                printf("\n");
                printf("\n");
                goto m2;
            }
```

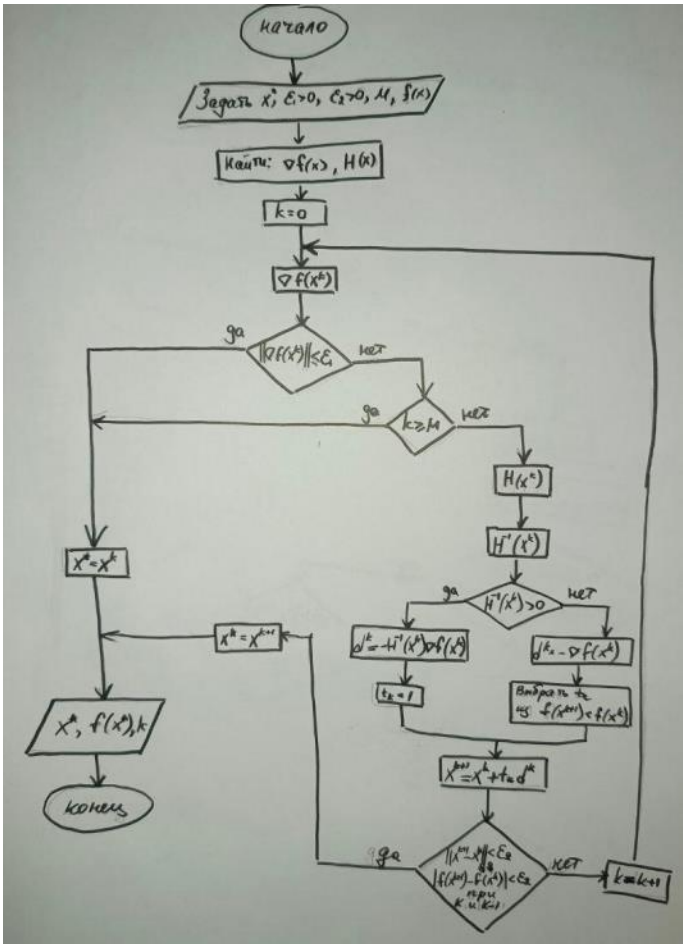
Результат:

```
Введите коэфиценты перед x1^2 x1x2 x2^2
2
0.2
6
Уровнение =   2.0*x^2 0.2*x1x2 6.0*x2^2
Введите x0 e1 e2 M
1.5
0.5
0.15
0.2
10
k = 0  ------  f(x)=[6.100000;6.300000]   xk= [1.500000;0.500000]   -----------------
Градиент xk > e1
Вычислим ||f(x)|| > e =   8.769265 > 0.150000
hess[0][0] = 0.250209 hess[0][1]= -0.004170 hess[1][0]= -0.004170 hess[1][1] = 0.083403
Проверим на выполение обоих условий ||xk+1 - xk ||= 1.581667 > 0.200000  ------   |f(xk+1) -  f(xk) |=6.149995 > 0.200000
Условие не выполнено



k = 1  ------  f(x)=[-0.003169;0.009842]   xk= [-0.000834;0.000834]   -----------------
Градиент xk < e1
Ответ x* = [-0.000834;0.000834]     f(x*)=0.000005
```

Блок схема метода



Вывод: была описана работа метода Ньютона, также по этому алгоритму по поиску минимума функции была реализована программа на языке высокого уровня C++.



начало
Задать x^0, $\varepsilon_1>0$, $\varepsilon_2>0$, M, $f(x)$
Найти: $\nabla f(x)$, $H(x)$
$k=0$
$\nabla f(x^k)$
да
$\|\nabla f(x^k)\| \le \varepsilon_1$
нет
да
$k \ge M$
нет
$H(x^k)$
$H^{-1}(x^k)$
да
$H^{-1}(x^k)>0$
нет
$d^k=-H^{-1}(x^k)\nabla f(x^k)$
$d^k=-\nabla f(x^k)$
$t_k=1$
Выбрать t_k из: $f(x^{k+1})<f(x^k)$
$x^{k+1}=x^k+t_k d^k$
$x^*=x^k$
$x^*=x^{k+1}$
x^*, $f(x^*)$, k
конец
да
$\|x^{k+1}-x^k\|<\varepsilon_2$
$|f(x^{k+1})-f(x^k)|<\varepsilon_2$
при k и $k-1$
нет
$k=k+1$