Московский государственный технический университет

им. Н.Э. Баумана

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Лабораторная работа № 3 по курсу

«Методы вычислений»

**«Метод парабол»**

Работу выполнил

студент группы ИУ7-23М

Кадыров Руслан

Москва, 2018г.

**Цель работы**

Изучение метода парабол для решения задачи одномерной минимизации.

**Постановка задачи**

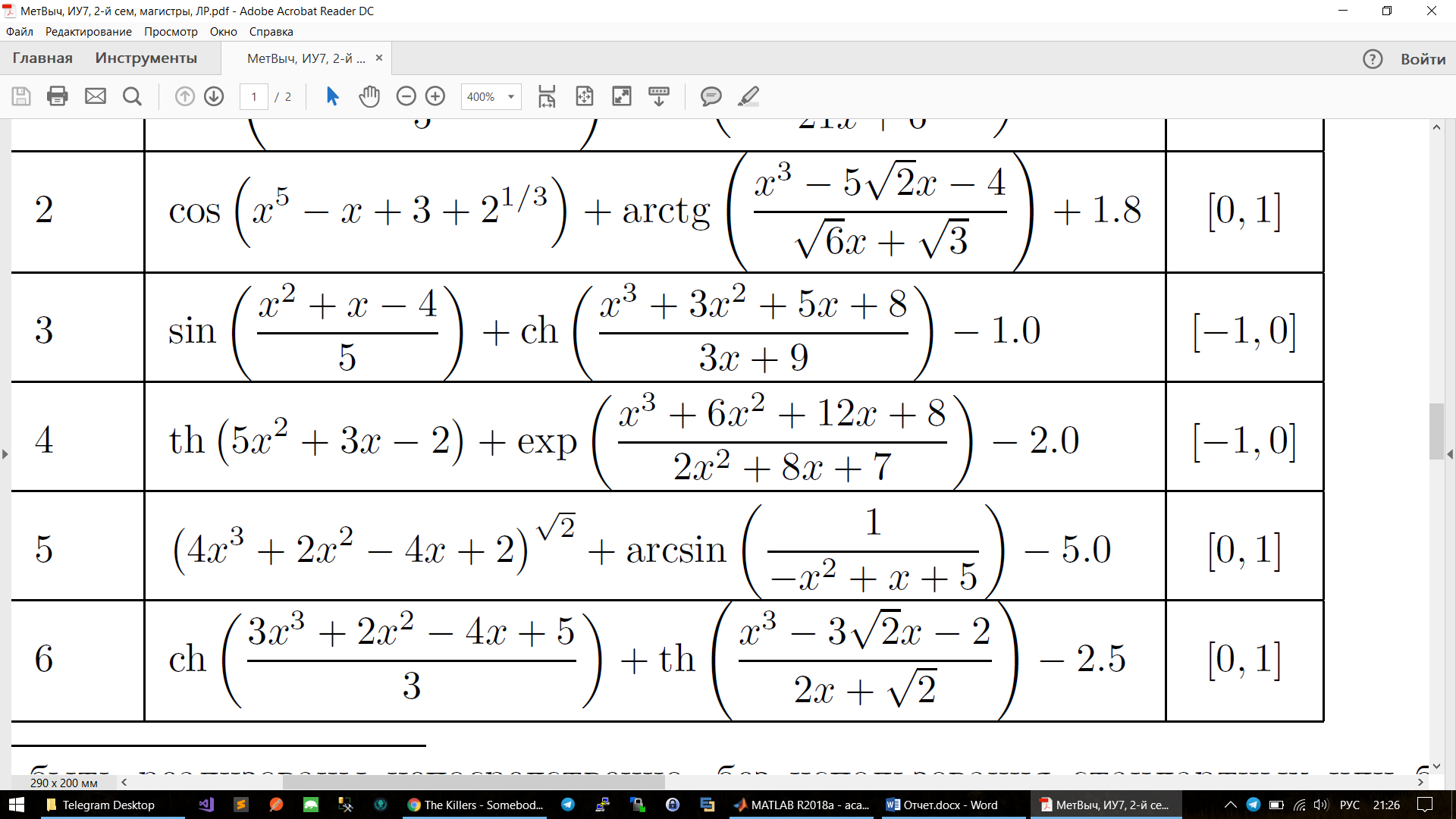
1. Реализовать метод парабол в сочетании с методом парабол в виде программы на ЭВМ;
2. Провести решение задачи

для индивидуального варианта;

1. Организовать вывод на экран графика целевой функции, найденной точки минимума (*x\**, *f* (*x\**)) и последовательности точек (*xi*, *f* (*xi*)), приближающих точку искомого минимума.

Исходные данные задачи: вариант №6

Целевая функция



на отрезке [0,1];

**Описание метода парабол**

В методе парабол предлагается аппроксимировать оптимизируемую функцию *f(x)* с помощью квадратичной функции *p(x) = ax2 + bx + c*. Выбираются три точки *x1 <x2 <x3* такие, что интервал *[x1, x3]* содержит точку минимума функции *f* и *f(x1) ≥ f(x2) ≥ f(x3)*. Далее строится квадратный трехчлен, график которого проходит через три выбранные точки графика функции *f(x).* Если хотя бы одно из неравенств является строгим, то поиск *x\** окончен, так как функция является унимодальной. Приведем алгоритм метода парабол.

**Шаг 1.** Выбрать точки *x1, x2, x3* такие, что *x1 <x2 <x3* *f* и *f(x1) ≥ f(x2) ≥ f(x3)*. Берем в качестве исходные точек *x2 = x0, x1 = x0 – h, x3 = x0 + h, h* – малое число.

**Шаг 2.** Вычислить значение x по формуле

x^{k+1} = x^k - 0.5h\frac{f(x^k + h) - f(x^k - h)}{f(x^k + h) - 2f(x^k) + f(x^k - h)}

**Шаг 3.** Проверка на окончание поиска. Сравнить модуль разно­сти значений *х* на данной и предыдущей итерациях с *ε*. Если |*∆*|≤ *ε*, то вычисления завершить. Положить *x\* = x, f\* = f(x\*)*, иначе перейти к шагу 4.

**Шаг 4.** Выбрать новую тройку точек *x1, x2, x3* такие, что *x2 = x,* найденному ранее, а *x1 = x2 – h, x3 = x2 + h, h* – малое число.

**Текст программы**

Листинг скрипта MetodyVich\_Lab\_3.m

eps = 0.01;

a = 0;

b = 1;

[xRes, fRes, xi, fi, iterCount] = ParabolasMethod(a,b, eps);

%Получение данных для построения графика целевой функции

xArr = zeros(1,iterCount);

fArr = zeros(1,iterCount);

a = 0;

step = 1;

while(step < 600)

xArr(step) = step\*0.01 + a;

fArr(step) = Func(step\*0.01 + a);

step = step + 1;

end

%Получение данных для построения точек, приближающихся к минимуму

xI\_Arr = zeros(1,iterCount);

fI\_Arr = zeros(1,iterCount);

step = 1;

for i=1:MaxIterationCount()

if(xi(i)~=0)

xI\_Arr(step) = xi(i);

fI\_Arr(step) = fi(i);

step = step + 1;

end

end

fprintf('Количество вычислений целевой функции: %1.d.\n',iterCount);

fprintf('x\* = %1.10f.\n',xRes);

fprintf('f(x\*) = %1.10f.\n',fRes);

plot(xArr, fArr,xRes, fRes, 'ro',xI\_Arr,fI\_Arr,'k-\*');

grid on;

title('График целевой функции f(x)');

xlabel('x');

ylabel('Значение целевой функции f(x)');

ylim([-2 15])

%Вычисление значения целевой функции в точке х

function X = Func(x)

X = cosh((3\*(x^3)+2\*(x^2)-4\*x+5)/3)+tanh((x^3-3\*sqrt(2)\*x-2)/(2\*x+sqrt(2)))-2.5;

end

%Метод парабол

function [xResult, fResult, xI,fI, iterationCount] = ParabolasMethod(a, b, eps)

% a - начало отрезка, b - конец отрезка, eps - точность поиска

% xResult - оптимальный x\*, %fResult - значение целевой функции в x\*

% xI-последовательность xi, приближающих точку искомого минимума

% fI-последовательность fi, приближающих точку искомого минимума

% iterationCount - число вычислений значения целевой функции

xi = zeros(1,MaxIterationCount());

fi = zeros(1,MaxIterationCount());

xCenter = (b - a)/2;

step = 0.001\*xCenter;

xMin = 0;

iter = 1;

pointsCount = 1;

while(true)

xLeft = xCenter - step;

xRight = xCenter + step;

xi(pointsCount) = xLeft;

xi(pointsCount + 1) = xRight;

fi(pointsCount) = Func(xLeft);

fi(pointsCount + 1) = Func(xRight);

pointsCount = pointsCount + 2;

xMin = 0.5\*(Func(xLeft)\*(xRight + xCenter) - 2\*(Func(xCenter)\*(xRight + xLeft)) + Func(xRight)\*(xCenter + xLeft))/(Func(xLeft) - 2\*Func(xCenter) + Func(xRight));

iter = iter + 1;

if(abs(xMin - xCenter) < eps)

break;

end

xCenter = xMin;

end

iterationCount = iter - 1;

xI = xi;

fI = fi;

xResult = xMin;

fResult = Func(xResult);

end

function num = MaxIterationCount

num = 250;

end

**Результаты расчета задачи индивидуального варианта**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | ***ε*** | *N* | *x\** | *f (x\*)* |
| 1 | 10-2 | 2 | 0.4824182917 | -1.4738932844 |
| 2 | 10-4 | 3 | 0.4824182540 | -1.4738932844 |
| 3 | 10-6 | 3 | 0.4824182540 | -1.4738932844 |