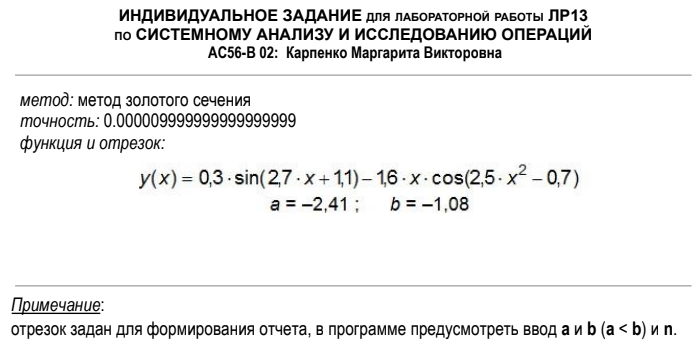
Карпенко Маргарита, АС-56

Лабораторная работа №13



*Код программы:*

#include <iostream>

#include <iomanip>

using namespace std;

double function(double x)

{

return 0.3 \* sin(2.7 \* x + 1.1) - 1.6 \* x \* cos(2.5 \* x \* x - 0.7);

}

double GoldenRatioMax(double E, double alfa, double a,double b)

{

double x1 = b - (b - a) \* alfa;

double x2 = a + (b - a) \* alfa;

while (abs(b - a) >= E)

{

if (function(x1) > function(x2))

{

b = x2;

x2 = x1;

x1 = b - (b - a) \* alfa;

}

else

{

a = x1;

x1 = x2;

x2 = a + (b - a) \* alfa;

}

}

return (a + b) / 2;

}

double GoldenRatioMin(double E, double alfa, double a, double b)

{

double x1 = b - (b - a) \* alfa;

double x2 = a + (b - a) \* alfa;

while (abs(b - a) >= E)

{

if (function(x1) < function(x2))

{

b = x2;

x2 = x1;

x1 = b - (b - a) \* alfa;

}

else

{

a = x1;

x1 = x2;

x2 = a + (b - a) \* alfa;

}

}

return (a + b) / 2;

}

void solution()

{

double a, b, n, h;

double function\_values[2][22];

cout << "Table of function:\n" << endl;

cout << "a = ";

cin >> a;

cout << "b = ";

cin >> b;

cout << "n = ";

cin >> n;

h = (a + b \* -1) / n;

if (h < 0)

h = h \* -1;

cout << "h = " << h << endl;

cout << "----------------------" << endl;

cout << "\tx :\ty" << endl;

cout << "----------------------" << endl;

double xx;

double x = a;

for (int i = 0; i < n + 1; i++)

{

xx = x + i \* h;

if (xx >= 0)

cout << " ";

cout << fixed << setprecision(5) << xx << " : ";

function\_values[0][i] = xx;

if (function(xx) >= 0)

cout << " ";

cout << fixed << setprecision(5) << function(xx) << endl;

function\_values[1][i] = function(xx);

}

cout << "----------------------" << endl << endl;

double alfa = 0.61803;

double E = 0.000009999999999999999;

double extremums[4][4] = { 0 };

string extremum[4];

int k = 0;

for (int j = 1; j < n - 1; j++)

{

if (function\_values[1][j] > function\_values[1][j - 1] && function\_values[1][j] > function\_values[1][j + 1])

{

extremums[k][0] = function\_values[0][j - 1];

extremums[k][1] = function\_values[0][j + 1];

extremum[k] = "max";

extremums[k][2] = GoldenRatioMax(E, alfa, extremums[k][0], extremums[k][1]);

extremums[k][3] = function(extremums[k][2]);

k++;

}

else if (function\_values[1][j] < function\_values[1][j - 1] && function\_values[1][j] < function\_values[1][j + 1])

{

extremums[k][0] = function\_values[0][j - 1];

extremums[k][1] = function\_values[0][j + 1];

extremum[k] = "min";

extremums[k][2] = GoldenRatioMin(E, alfa, extremums[k][0], extremums[k][1]);

extremums[k][3] = function(extremums[k][2]);

k++;

}

}

cout << "Research of function (exp = 0.000009999999999999999)" << endl;

cout << "--------------------------------------------------------" << endl;

cout << " x-h : x+h : type : x\_extr : y\_extr " << endl;

cout << "--------------------------------------------------------" << endl;

for (int i = 0; i < 4; i++)

{

for (int j = 0; j < 4; j++)

{

if (j == 2)

{

cout << extremum[i] << " : ";

}

if (extremums[i][j] > 0)

cout << " ";

if (j == 3)

{

cout << extremums[i][j];

continue;

}

cout << extremums[i][j] << " : ";

}

cout << endl;

}

}

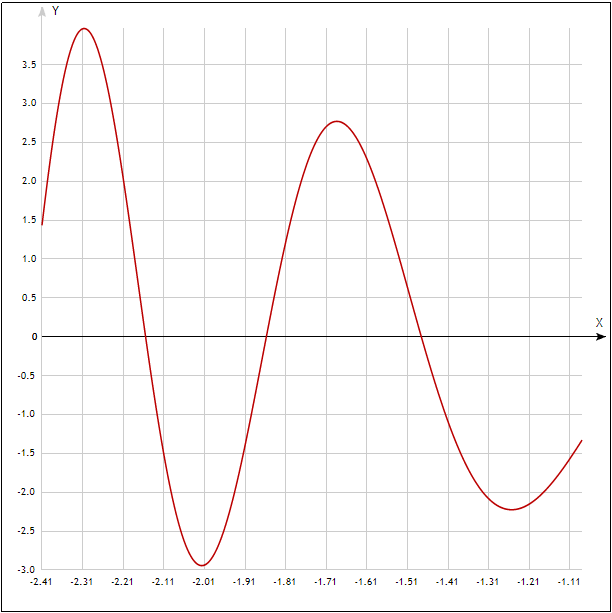
int main()

{

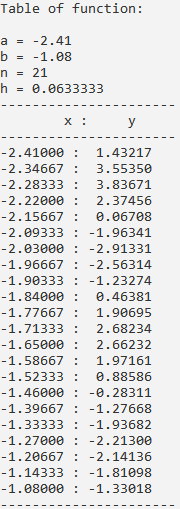
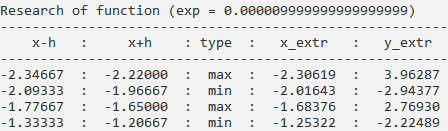
solution();

}

*График функции:*



*Результат работы программы:*

**