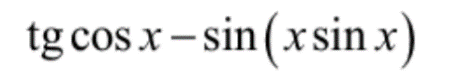
Практическая работа №5

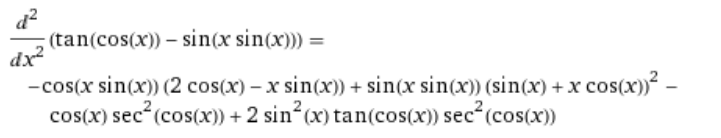
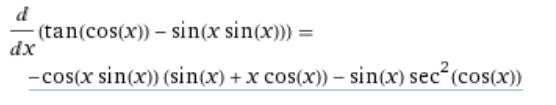
по дисциплине «Вычислительная математика»

**«Численное дифференцирование»**

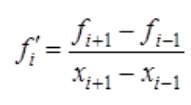
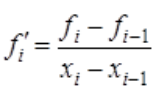
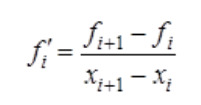
студента группы M3207 Бойцова Виталия

f(x) =

Равноотстоящие узлы:



Для первого узла: Для последнего: Для остальных узлов:



//Sometimes OOP isn't usefull (C) Vitaly Boytsov

#include <iostream>

#include <vector>

#include <cmath>

#include <algorithm>

using namespace *std*;

const double EPS = 1e-8;

const double PI = *acos*(-1);

*vector*<double> calculateDerivate(*vector*<double> y, double h) {

*vector*<double> ans(y.*size*());

ans.*front*() = (y[1] - y[0]) / h;

for (int i = 1; i < (int)y.*size*() - 1; i++)

ans[i] = (y[i + 1] - y[i - 1]) / (2 \* h);

ans.*back*() = (y[y.*size*() - 1] - y[y.*size*() - 2]) / h;

return ans;

}

double sec(double x) {

return 1 / *cos*(x);

}

int main() {

int n;

double a, b, h;

*cin* >> a >> b >> n;

h = (b - a) / n;

//находим узлы

*vector*<double> arr(n), values(n);

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

double x = arr[i] = a + i \* (b - a) / (n - 1);

values[i] = *tan*(*cos*(x)) - *sin*(x \* *sin*(x));

}

*vector*<double> apr1(n), apr2(n);

//approximate derivate

apr1 = calculateDerivate(values, h);

apr2 = calculateDerivate(apr1, h);

//считаем значения производной в точках аналитически и находим погрешности

double d1 = 0, d2 = 0;

for(int i = 0; i < arr.*size*(); i++) {

double x = arr[i];

//analytic derivate

double an1 = -*sin*(x) \* *pow*(sec(*cos*(x)), 2)

- *cos*(x \* *sin*(x)) \* (x \* *cos*(x) + *sin*(x));

double an2 = -(*cos*(x) \* *pow*(sec(*cos*(x)), 2))

- *cos*(x \* *sin*(x)) \* (2 \* *cos*(x) - x \* *sin*(x))

+ *pow*(x \* *cos*(x) + *sin*(x), 2) \* *sin*(x \* *sin*(x))

+ 2 \* *pow*(sec(*cos*(x)), 2) \* *pow*(*sin*(x), 2) \* *tan*(*cos*(x));

double ap1 = apr1[i];

double ap2 = apr2[i];

*printf*("%lf %lf %lf %lf %lf\n", x, an1, ap1, an2, ap2);

d1 = *max*(d1, *abs*(an1 - ap1));

d2 = *max*(d2, *abs*(an2 - ap2));

}

*cout* << d1 << ' ' << d2 << '\n';

}