

```

#include<iostream>
#include<vector>
#include<assert.h>
#include<algorithm>
#include<string>
#include <windows.h>
using namespace std;

double f1(double y) // Подсчёт значения y в точке x
{
    return cos(y + 0.5) - 2;
}
double f2(double x)
{
    return (sin(x) - 1) / 2;
}
double F1(double x, double y)
{
    return cos(y + 0.5) - x - 2;
}
double F2(double x, double y)
{
    return sin(x) - 2 * y - 1;
}
double dx(double x, double y, double dy)
{
    return cos(y + 0.5) - sin(y + 0.5) * dy - x - 2;
}
double dy(double x, double y)
{
    return (cos(x) * cos(y + 0.5) - cos(x) * x - 2 * cos(x) + sin(x) - 2 * y - 1) / (sin(y
+ 0.5) * cos(x) + 2);
}
int main()
{
    double x, xi, y, yi, e, X, Y;
    SetConsoleOutputCP(CP_UTF8);
    cout << "Введите начальное приближение x и y, а так же точность e: ";
    cin >> x >> y >> e;
    X = x;
    Y = y;
    cout << "\n\nМетод простых итераций:";
    do // Производная от обеих функций, а так же построенный график смотреть в папке
предмета
    {
        xi = x;
        yi = y;
        x = f1(y);
        y = f2(xi);
    } while (abs(y - yi) > e && abs(x - xi) > e);
    cout.precision();
    cout << "\nX = " << fixed << x << endl << fixed << "Y = " << y;
    x = X;
    y = Y;
    cout << "\nМетод Ньютона:";
    do
    {
        yi = dy(x, y);
        xi = dx(x, y, yi);
        x = x + xi;
        y = y + yi;
    } while (!(F1(x + e, y) > 0 && F1(x - e, y) < 0) || (F1(x + e, y) < 0 && F1(x - e, y)
> 0)) && !(F1(x, y + e) > 0 && F1(x, y - e) < 0) || (F1(x, y - e) < 0 && F1(x, y + e) > 0));
    cout << "\nX = " << x << endl << "Y = " << y;
}

```