

```

#include <bits/stdc++.h>
using namespace std;
float dydx(float x, float y) {
    return (x - y);
}
float exactSolution(float x) {
    return x - 1 + exp(-x);
}
float rungeKutta(float x0, float y0, float x, float h) {
    int n = (int)((x - x0) / h);
    float k1, k2, k3, k4;

    float y = y0;
    for (int i = 1; i <= n; i++) {
        k1 = h * dydx(x0, y); k2 = h * dydx(x0 + 0.5 * h, y + 0.5 * k1);
        k3 = h * dydx(x0 + 0.5 * h, y + 0.5 * k2); k4 = h * dydx(x0 + h, y + k3);
        y = y + (1.0 / 6.0) * (k1 + 2 * k2 + 2 * k3 + k4); x0 = x0 + h;
    }
    return y;
}
int main() {
    float x0 = 0, y0 = 1, x = 10, h = 0.1;
    float meanError = 0.0;
    int steps = (int)((x - x0) / h);
    cout << "X\t\tY\t\tYe\t\tabs(Y - Ye)\n";
    for (int i = 0; i <= steps; i++) {
        float currentX = x0 + i * h; float Y = rungeKutta(x0, y0, currentX, h);
        float Ye = exactSolution(currentX);
        float error = abs(Y - Ye);
        cout << fixed << setprecision(6) << currentX << "\t" << Y << "\t" << Ye << "\t" << error << "\n";
        meanError += error;
    }
    meanError /= (steps + 1);
    cout << "Mean Error: " << fixed << setprecision(6) << meanError << "\n";
    return 0;
}

```