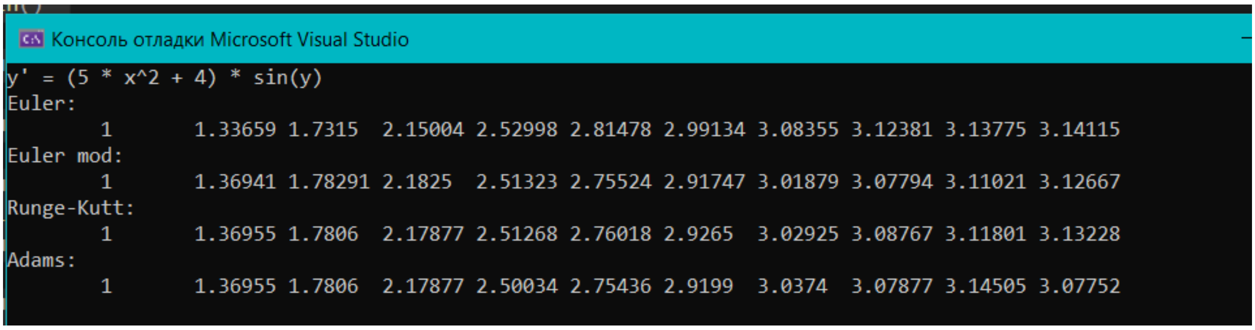
Фролова Анастасия, ВМК-20

Вычмат, 12 вариант

```
#include <iostream>
#include <locale>
#include <vector>
#include <string>
#include <iomanip>
using namespace std;

//функция
double f(double x, double y)
{
    return (6 * pow(x, 2) +4 ) * sin(y);
}

//метод Эйлера
void euler(double x0, double y0, double h, double e)
{
    double x = x0, y = y0;

    for (int i = 0; i < 11; i++)
    {
        cout << "\t" << y;

        y = y + h * f(x, y);

        x = x + h;
    }
}

//модифицированный метод эйлера
void euler_modif(double x0, double y0, double h, double e)
{
    double x = x0, y = y0;

    for (int i = 0; i < 11; i++)
    {
        cout << "\t" << y;

        y = y + h * f(x + h / 2, y + h / 2 * f(x, y));

        x = x + h;
    }
}

//метод Рунге-Кутта
void runge_kutt(double x0, double y0, double h, double e)
{
    vector<double> z(4);

    double x = x0, y = y0;
    for (int i = 0; i < 11; i++)
    {
        //вычисляем коэффициенты
        z[0] = h * f(x, y);
        z[1] = h * f(x + h / 2, y + z[0] / 2);
        z[2] = h * f(x + h / 2, y + z[1] / 2);
        z[3] = h * f(x + h, y + z[2]);

        cout << "\t" << y;

        y = y + (z[0] + 2 * z[1] + 2 * z[2] + z[3]) / 6;

        x = x + h;
    }
}

//метод Адамса
void adams(double x0, double y0, double h, double e)
{
    vector<double> z(4), yk(4), xk(4);

    double x = x0, y = y0;
    double t = 0;

    yk[0] = y0;
    xk[0] = x0;

    cout << setw(9) << y0;

    for (int i = 1; i < 11; i++)
    {
        if (i < 4)
        {
            z[0] = h * f(x, y);
            z[1] = h * f(x + h / 2, y + z[0] / 2);
            z[2] = h * f(x + h / 2, y + z[1] / 2);
            z[3] = h * f(x + h, y + z[2]);

            y = y + (z[0] + 2 * z[1] + 2 * z[2] + z[3]) / 6;
            x = x + h;

            yk[i] = y;
            xk[i] = x;
        }
        else
        {
```

```
                    t = 55 * f(xk[3], yk[3]) - 59 * f(xk[2], yk[2]) + 37 * f(xk[1], yk[1]) - 9 * f(xk[0], yk[0]);
                    y = y + (h / 24) * t;

                    xk[0] = xk[0] + h;
                    xk[1] = xk[1] + h;
                    xk[2] = xk[2] + h;
                    xk[3] = xk[3] + h;

                    yk[0] = yk[1];
                    yk[1] = yk[2];
                    yk[2] = yk[3];
                    yk[3] = y;
                }
            cout << "\t" << y;
        }
        return;
    }
    int main()
    {
        double x0 = 0, y0 = 1, h = 0.1, e = 0.001;

        cout << "y' = (5 * x^2 + 4) * sin(y)" << endl;

        cout << "Euler:" << endl;
        euler(x0, y0, h, e);
        cout << endl;

        cout << "Euler mod:" << endl;
        euler_modif(x0, y0, h, e);
        cout << endl;

        cout << "Runge-Kutt:" << endl;
        runge_kutt(x0, y0, h, e);
        cout << endl;

        cout << "Adams:" << endl;
        adams(x0, y0, h, e);
        cout << endl;

        return 0;
    }
```



```
y' = (5 * x^2 + 4) * sin(y)
Euler:
        1       1.33659 1.7315  2.15004 2.52998 2.81478 2.99134 3.08355 3.12381 3.13775 3.14115
Euler mod:
        1       1.36941 1.78291 2.1825  2.51323 2.75524 2.91747 3.01879 3.07794 3.11021 3.12667
Runge-Kutt:
        1       1.36955 1.7806  2.17877 2.51268 2.76018 2.9265  3.02925 3.08767 3.11801 3.13228
Adams:
        1       1.36955 1.7806  2.17877 2.50034 2.75436 2.9199  3.0374  3.07877 3.14505 3.07752
```