

**Решение:**

****

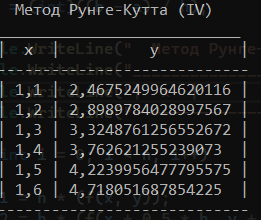
1. Найдем шаг интегрирования для решения задачи Коши методом Рунге–Кутта (IV) с точностью 10−4

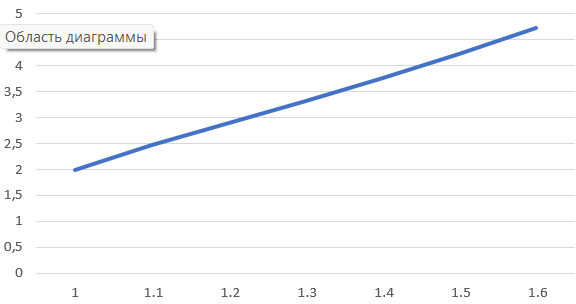
y’=

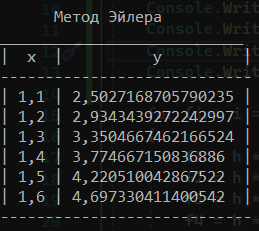
h0==0,1

Выберем x0 = a = 1, y0 = 2. Найдем решение данной задачи Коши методом Рунге–Кутта (IV)

1. **Скриншоты:**

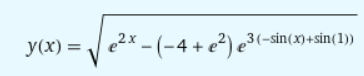
****

1. Построим приближенную интегральную кривую, полученную методом Рунге–Кутта (IV). 
2. Найдем решение задачи Коши на отрезке [a, b] методом Эйлера

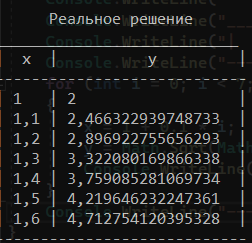


1. Найдем точное решение задачи Коши.

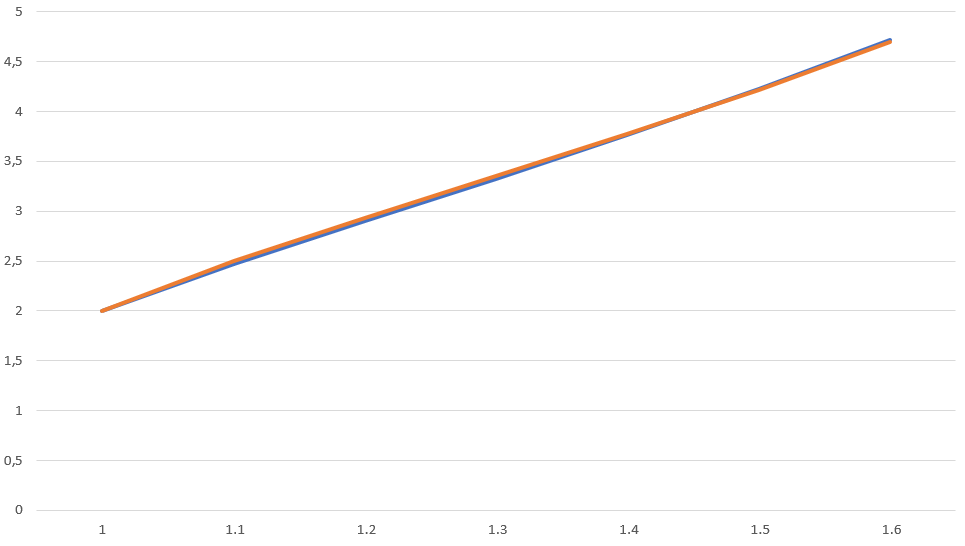
y’=

Его решение будет выглядеть как 

Значения у при разных х



1. Построим на одном графике две интегральных кривых



1. **Код:**

double x = 1, y = 2, a = 1, b = 1.6, h = 0.1;

rungeKutta(x, y, a, b, h);

euler(x, y, h);

real();

void rungeKutta(double x, double y, double a, double b, double h)

{

double f1, f2, f3, f4;

int n = (int)((b - a) / h);

Console.WriteLine(" Метод Рунге–Кутта (IV)");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("| x | y |");

Console.WriteLine("----------------------------");

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

{

f1 = h \* (f(x, y));

f2 = h \* (f(x + 0.5 \* h, y + 0.5 \* f1));

f3 = h \* (f(x + 0.5 \* h, y + 0.5 \* f2));

f4 = h \* (f(x + h, y + f3));

y += (1.0 / 6.0) \* (f1 + 2 \* f2 + 2 \* f3 + f4);

x += h;

Console.WriteLine("| {0,-3} | {1,-18} |", (float)x, y);

}

Console.WriteLine("----------------------------\n");

}

void euler(double x, double y, double h)

{

Console.WriteLine(" Метод Эйлера");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("| x | y |");

Console.WriteLine("----------------------------");

for (int i = 0; i < 6; i++)

{

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

x = x + h;

Console.WriteLine("| {0,-3} | {1,-18} |", (float)x, y);

}

Console.WriteLine("----------------------------\n");

}

static double f(double x, double y)

{

return Math.Pow(2.71, 2\*x) \* (2 + 3 \* Math.Cos(x)) / (2 \* y) - (3 \* y \* Math.Cos(x)) / 2;

}

void real()

{

double x = 0;

Console.WriteLine(" Реальное решение");

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("| x | y |");

Console.WriteLine("----------------------------");

for (int i = 0; i < 7; i++)

{

x = 1 + 0.1 \* i;

y = Math.Sqrt(Math.Pow(2.71, 2 \* x) - (-4 + 2.71 \* 2.71) \* Math.Pow(2.71, 3 \* (-Math.Sin(x)+Math.Sin(1))));

Console.WriteLine("| {0,-3} | {1,-18} |", (float)x, y);

}

Console.WriteLine("----------------------------");

}

**Вывод:** Метод Рунге-Кутта хоть и является более точным по сравнению с методом Эйлера и соответственно самым близким к решению уравнения найденному мной самим, но является так же и более трудоемким, а на моем интервале [1;1,6] отклонение графиков друг от друга минимально соответственно решение более трудоемкими методами в моем случае избыточно.