Лабораторная работа 1.

Вариант 3.

Решение дифференциальных уравнений.

Решить вручную методом Эйлера и написать программу, сравнить ручной расчёт с результатом работы программы.

Дано: a=1, b=2, y_0=3, n=10.

$$y' = \frac{3y - 7x}{4y - 3x};$$

$$C = 4y^2 - 6yx + 7x^2.$$

Выполнение.

1. Проверим, что функция, приведённая в последней строке задания, действительно является решением данного уравнения. Продифференцируем обе части её уравнения по x:

$$8yy' - 6y - 6y'x + 14x = 0;$$

$$4yy' - 3y'x - 3y + 7x = 0;$$

$$y'(4y - 3x) = 3y - 7x;$$

$$y' = \frac{3y - 7x}{4y - 3x}.$$

Получили значение производной, соответствующее заданному дифференциальному уравнению. Следовательно, функция является решением уравнения. По её значениям можно проверять правильность и точность вычислений.

2. *Ручное решение дифференциального уравнения методом Эйлера.*

Данное уравнение уже записано в стандартном виде, разрешённом относительно производной искомой функции:

$$\frac{dy}{dx} = f(x, y).$$

Поэтому для решаемого уравнения имеем

$$f(x, y) = \frac{3y - 7x}{4y - 3x}.$$

Будем вести расчёт для каждого узла заданной сетки, то есть, с шагом

$$h = \frac{b - a}{n} = \frac{2 - 1}{10} = 0{,}1.$$

Начальные значения нам известны: при x_0=a=1 y_0=3.

Каждое последующее значение функции y будем вычислять по формуле Эйлера:

$$x_i = x_{i-1} + h;$$
$$y_i = y_{i-1} + h \cdot f(x_{i-1}, y_{i-1}).$$

Сведём все вычисления по ручному решению задачи в таблицу:

i	x_i	y_i	$f(x_{i-1}, y_{i-1})$	$C_i = 4y_i^2 - 6y_ix_i + 7x_i^2$	$\Delta C = C_i - C_0$
0	1	3	0,22222	25	0
1	1,1	3,02222	0,1555	25,0586	0,0586
2	1,2	3,03777	0,08342	25,12024	0,12024
3	1,3	3,04611	0,00463	25,18549	0,18549
4	1,4	3,04657	-0,08268	25,25517	0,25517
5	1,5	3,0383	-0,18098	25,33037	0,33037
6	1,6	3,0202	-0,29384	25,41251	0,41251
7	1,7	2,99082	-0,42655	25,50365	0,50365
8	1,8	2,94816	-0,58748	25,60646	0,60646
9	1,9	2,88941	-0,79072	25,72549	0,72549
10	2	2,81034	-1,06251	25,86796	0,86796

Из последней колонки видно, что погрешность вычислений накапливается с каждым шагом.

2. *Численное решение с помощью программы.*

Реализуем расчёт в книге Excel.

На первой строке листа впишем границы интервала, начальное значение y и требуемое количество интервалов. Программа будет использовать эти значения при вычислениях.

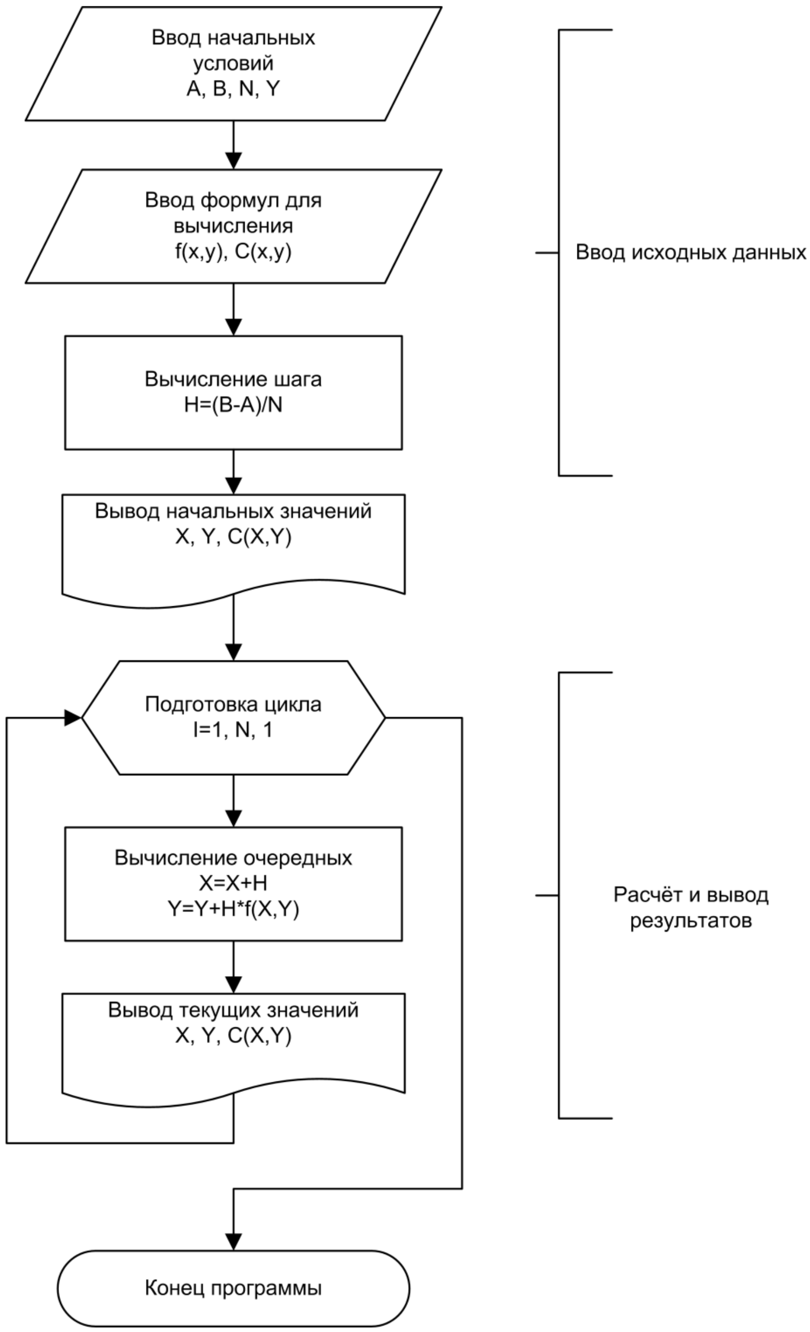
Создадим две макрокоманды, которые назначим двум клавишам. Первая из них будет очищать содержимое области расчёта, вторая – выполнять непосредственно расчёт. При этом вручную очищать область необязательно, так как перед расчётом очистка будет производиться программно.

Выполнение расчёта будет делаться в той же последовательности, что и вручную. Сначала в переменные считываются исходные данные, вычисляется размер шага H, текущим значениям X и Y присваиваются начальные значения, а затем в цикле будут вычислены и выведены все остальные числовые величины. Цикл прекратит выполнение, когда счётчик циклов достигнет заданного по условию количества шагов. Это количество не задаётся в программе жёстко, а вводится в ячейку рабочего листа.

Округление результатов при выводе производится до пятого знака.

Блок-схема и листинг разработанных макросов приводятся ниже.

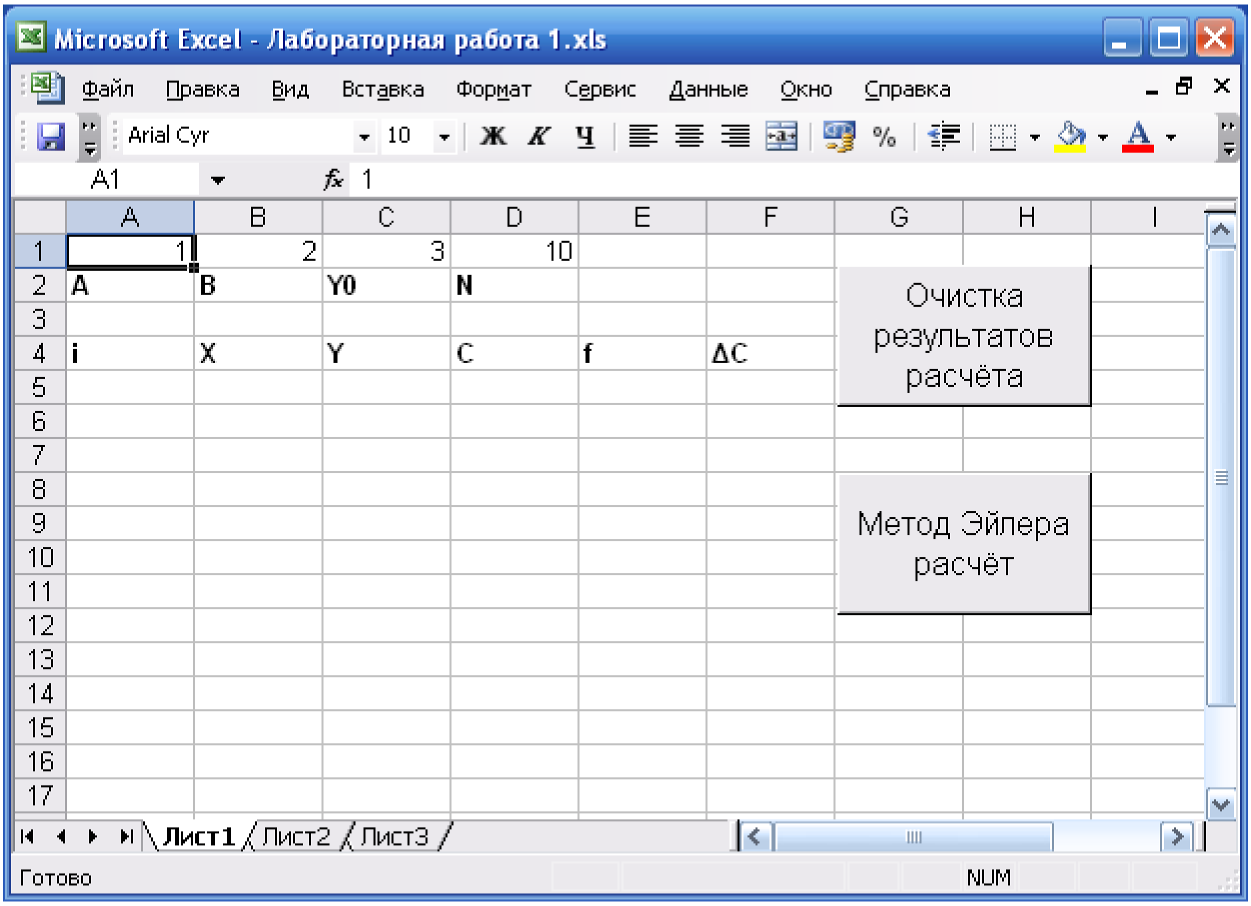
Блок схема расчёта по методу Эйлера.



Листинг программы.

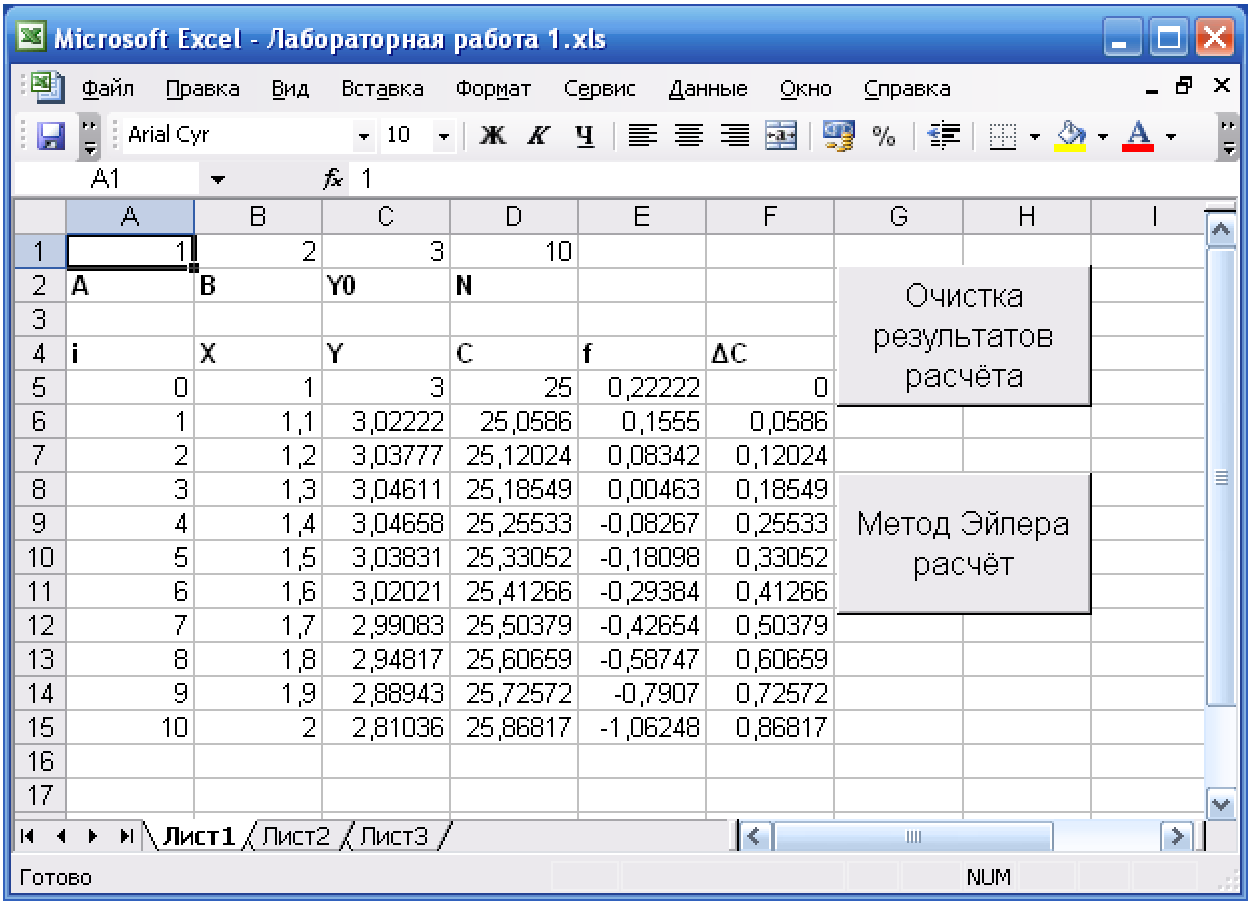
```
Sub Макрос1()
' Макрос1 Макрос
' Макрос записан 25.12.2014
' Очистка области расчёта.

    Range("A5").Select
    If ActiveCell.SpecialCells(xlLastCell).Row>=5 Then
     Range(Selection, Cells(ActiveCell.SpecialCells(xlLastCell).Row, 6)).Select
     Selection.ClearContents
    End If
    Range("A1").Select

End Sub

Sub Макрос2()
' Макрос2 Макрос
' Макрос записан 25.12.2014
' Выполнение расчёта по методу Эйлера.

' Вызываем макрос очистки области расчёта
Макрос1

' Считываем исходные значения в переменные.
A = Range("A1")
B = Range("B1")
Y0 = Range("C1")
N = Range("D1")

' Рассчитываем размер интервала
H = (B - A) / N
Range("K1") = H

' Проходим по строкам, вычисляем и выводим значение каждой ячейки
For i = 0 To N

 ' j - номер текущей строки - вспомогательная переменная
 j = 5 + i
 Cells(j, 1) = i

 ' Вычисляем и выводим текущие значения X и Y (первые значения - начальные)
 If i = 0 Then
  X = A
  Y = Y0
 Else
  X = X + H
  Y = Y + H * f
 End If
 Cells(j, 2) = Round(X, 5)
 Cells(j, 3) = Round(Y, 5)

 ' Текущее значение C
 C = 4 * Cells(j, 3) ^ 2 - 6 * Cells(j, 3) * Cells(j, 2) + 7 * Cells(j, 2) ^ 2
 Cells(j, 4) = Round(C, 5)

 ' Текущее значение f
 f = (3 * Cells(j, 3) - 7 * Cells(j, 2)) / (4 * Cells(j, 3) - 3 * Cells(j, 2))
 Cells(j, 5) = Round(f, 5)

 ' отклонение вычисленной константы C от исходной
 Cells(j, 6) = Round(C - Range("D5"), 5)

Next

End Sub
```

Копия экрана после выполнения очистки области результатов:



	A	B	C	D	E	F
1	1	2	3	10		
2	A	B	Y0	N		
3						
4	i	X	Y	C	f	ΔC

Копия экрана после выполнения расчёта:



	A	B	C	D	E	F
1	1	2	3	10		
2	A	B	Y0	N		
3						
4	i	X	Y	C	f	ΔC
5	0	1	3	25	0,22222	0
6	1	1,1	3,02222	25,0586	0,1555	0,0586
7	2	1,2	3,03777	25,12024	0,08342	0,12024
8	3	1,3	3,04611	25,18549	0,00463	0,18549
9	4	1,4	3,04658	25,25533	-0,08267	0,25533
10	5	1,5	3,03831	25,33052	-0,18098	0,33052
11	6	1,6	3,02021	25,41266	-0,29384	0,41266
12	7	1,7	2,99083	25,50379	-0,42654	0,50379
13	8	1,8	2,94817	25,60659	-0,58747	0,60659
14	9	1,9	2,88943	25,72572	-0,7907	0,72572
15	10	2	2,81036	25,86817	-1,06248	0,86817

3. Для сравнения результатов ручного вычисления и программного расчёта приведём таблицу расхождений вычисленной константы C:

i	x_i	$\Delta C=C_i-C_0$ ручн.	$\Delta C=C_i-C_0$ прогр.
0	1	0	0
1	1,1	0,0586	0,0586
2	1,2	0,12024	0,12024
3	1,3	0,18549	0,18549
4	1,4	0,25517	0,25533
5	1,5	0,33037	0,33052
6	1,6	0,41251	0,41266
7	1,7	0,50365	0,50379
8	1,8	0,60646	0,60659
9	1,9	0,72549	0,72572
10	2	0,86796	0,86817

Значения отличаются, так как при работе программы промежуточные результаты хранились с высокой точностью, и только при выводе округлялись, а при ручном расчёте каждый раз округлялись до пяти значащих цифр после запятой. Однако, как оказалось, программный расчёт дал несколько меньшую точность.

Если увеличить количество шагов исходной сетки на заданном интервале, то точность значительно увеличится. Однако для ручного способа намного возрастёт объём и время выполняемой работы, повысится вероятность возникновения ошибок. Для программной реализации эти изменения будут практически незаметны.