# Лабораторная работа №5.

# Тема работы: Разработка и отладка циклических алгоритмов и программ с заданным числом повторений

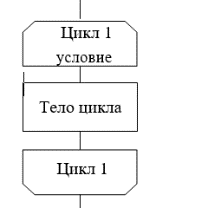
**Цель работы:** Сформировать умения разрабатывать и тестировать простые циклические алгоритмы и программы с заданным числом повторений, выполнять их отладку.

**Оснащение рабочего места:** ПК, VS, методические указания для проведения лабораторных работ, индивидуальные задания на ЛР

**Краткие теоретические сведения**

Иногда необходимо повторять одно и то же действие несколько раз подряд. Для этого используют циклы. Итерационные циклы — это циклы с заранее неизвестным числом шагов. Различают циклы с пред и пост условием. Графически цикл с предусловием представлен на рисунке 1.

Цикл с предусловием будет выполняться, пока условие, указанное в круглых скобках будет истинным. Синтаксис цикла с предусловием

while (Условие)

{

тело цикла;

}

Синтаксис цикла с постусловием: // форма записи оператора цикла do while:

do // начало цикла do while

{

/\*блок операторов\*/;

}

while (/\*условие выполнения цикла\*/); // конец цикла do while

Цикл с постусловием do while отличается от цикла while тем, что в do while сначала выполняется тело цикла, а затем проверяется условие продолжения цикла. Из-за такой особенности do while называют циклом с постусловием. Таким образом, если условие do while заведомо ложное, то хотя бы один раз блок операторов в теле цикла do while выполнится. В итоге do while отличается от цикла while структурой. Если в while сначала выполняется проверка условия продолжения цикла, и если условие истинно, то только тогда выполняется тело цикла. Цикл do while работает с точностью до наоборот, сначала выполняется тело цикла, а потом проверяется условие, вот почему тело цикла do while, хотя бы раз, выполнится. Графическое изображение цикла представлено на рисунке 2.

**Порядок выполнения работы**

1. Ознакомиться с краткими теоретическими сведениями
2. Ознакомиться с поставленной задачей на ЛР

# Задания

### Первый уровень сложности

Составить программу для определения таблицы значений функции ***у*** в произвольном диапазоне [***a***, ***b***] изменения аргумента ***х*** с произвольным шагом ***h***. Значения ***a*, *b*, *h*** вводятся с клавиатуры. Таблица должна содержать следующие столбцы: порядковый номер, значение аргумента ***x***, значение функции, сообщение о возрастании или убывании функции.

Определить максимальное и минимальное значения функции.

1.  *a* = –π; *b* = π; *h* = 0,4.

2.  *a* = 0,7; *b* = 1,8; *h* = 0,1.

3.  *a* = –0,5; *b* = 2,5; *h* = 0,2.

4.  *a* = –0,9; *b* = 2,7; *h* = 0,3.

5.  *a* = –2; *b* = 0,8; *h* = 0,2.

6.  *a* = –1,9; *b* = 2,7; *h* = 0,3.

7.  *a* = –0,4π; *b* = 0,4π; *h* = 0,5.

8.  *a* = –0,3π; *b* = 1,3π; *h* = π/10.

9.  *a* = –π/2; *b* = π/2; *h* = π/10.

10.  *a* = –3; *b* = 3; *h* = 0,5.

11.  *a* = –π; *b* = π; *h* = π/6.

12.  *a* = –0,9; *b* = 1, *h* = 0,3.

13.  *a* = –0,9; *b* = 2,7; *h* = 0,3.

14.  *a* = –0,1; *b* = 2; *h* = 0,1.

15.  *a* = π; *b* = 2π; *h* = π/15.

### Второй уровень сложности

Значение аргумента ***x*** изменяется от ***a*** до ***b*** с шагом ***h***. Для каждого ***x*** найти значения функции *Y*(*x*), суммы *S*(*x*) и |*Y*(*x*)–*S*(*x*)| и вывести в виде таблицы. Значения *a*, *b*, *h* и *n* вводятся программно. Так как значение *S*(*x*) является рядом разложения функции *Y*(*x*), значения *S* и *Y* для заданного аргумента *x* должны совпадать в целой части и в первых двух-четырех позициях после десятичной точки.

Работу программы проверить для *a* = 0,1; *b* = 1,0; *h* = 0,1; n=10.

1. , .

2.  .

3.   .

4.  .

5.  .

6.  .

7.  .

8. , .

9. , .

10.  , .

11. , .

12. , .

13. , .

14. , .

15. ****, **** .

**Содержание отчета:**

1. Титульный лист (см. образец отчёта)
2. Наименование и цель работы.
3. Задание на ЛР
4. Скопированный код из VS.
5. Скриншот выполненной программы
6. Ответы на контрольные вопросы.
7. Выводы по работе.

**Контрольные вопросы**

1. Какие операторы языка C++ используются для организации итерационных циклов?