

ЛАБОРАТОРНАЯ РАБОТА № 2

ЦИКЛИЧЕСКИЕ АЛГОРИТМЫ

1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью работы является получение обучающимися навыков реализации циклических алгоритмов на языке C++.

Задачами работы являются:

- написание программы на языке C++ с использованием различных типов циклов;
- тестирование работоспособности разработанной программы для различных исходных данных.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В программировании *цикл* – это повторяющееся выполнение одних и тех же простых или составных операторов. Алгоритмы, содержащие циклы, называются *циклическими*.

Существует несколько типов операторов цикла. Рассмотрим первый из них – оператор цикла с предусловием, или оператор цикла “пока”. На псевдокоде этот оператор записывается так:

```
пока условие  
    повторить  
        оператор
```

Составной оператор – это конструкция:

```
начало  
    оператор1  
    оператор2  
    ...  
    операторN
```

```
конец
```

Оператор (простой или составной), стоящий после служебного слова “повторить” и называемый *телом цикла*, будет выполняться циклически, пока выполняется *условие*, т. е. пока значение “условия” равно истине. Само условие цикла может быть логической константой, переменной или выражением с логическим результатом.

В языке C++ оператор цикла “пока” имеет вид:

```
while (условие) оператор
```

Конструкция, соответствующая циклу “пока”, приведена на рисунке 2.1а.

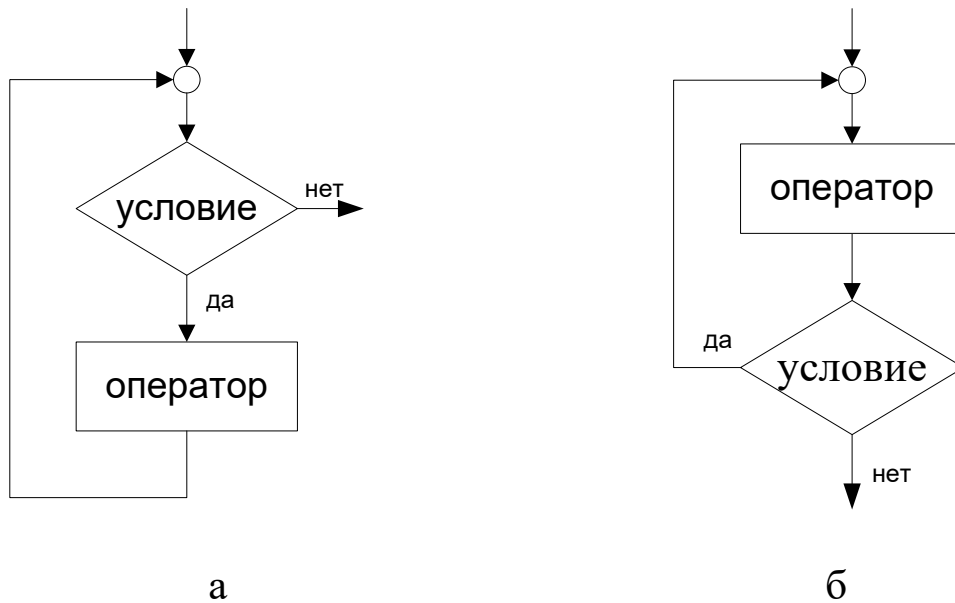


Рис. 2.1. Конструкция оператора цикла
а – с предусловием; б – с постусловием

Условие выполнения тела цикла “пока” проверяется до начала каждого его выполнения (итерации). Отсюда и название оператора – оператор *цикла с предусловием*. Если условие сразу не выполняется, то тело цикла игнорируется и будет выполняться оператор, стоящий сразу за телом цикла.

Рассмотренный оператор цикла с предусловием решает выполнять ли указанные действия до первой итерации. Если это не соответствует логике алгоритма, то можно использовать оператор *цикла с постусловием*, который решает выполнять или нет очередную итерацию после завершения предыдущей. Такой оператор называется оператором цикла “до”. Это имеет принципиальное значение лишь на первом шаге, а далее циклы ведут себя идентично. Цикл с постусловием на псевдокоде имеет вид:

```
повторить
    оператор
до    условие
```

В языке C++ оператору с постусловием соответствует оператор

```
do {
    операторы
}while (условие)
```

Оператор `do-while` называется оператором цикла с постусловием. Оператору `do-while` соответствует конструкция, представленная на рисунке 2.16.

Операторы циклов с пред- и постусловием, хотя и обладают значительной гибкостью, не слишком удобны для организации циклов, которые должны быть проделаны заданное число раз. Цикл с параметром вводится именно для таких случаев. В этом операторе для каждого значения параметра выполняется тело цикла (очередная итерация). Пределы изменения параметра и способ получения следующего значения параметра по текущему указываются в заголовке оператора цикла с параметром. На псевдокоде оператор цикла с параметром имеет вид:

для инициализация, условие, изменение
повторить
оператор

где “инициализация” используется для присвоения начального значения параметру цикла, “условие” определяет, выполнять тело цикла (оператор) или завершить цикл, “изменение” определяет способ изменения параметра цикла. Выполнение цикла происходит до тех пор, пока “условие” истинно. Как только условие становится ложным, начинает выполняться следующий за циклом оператор. Проверка “условия” происходит до выполнения тела цикла, т.е. этот оператор является частным случаем оператора цикла с предусловием. “Изменение” параметра цикла осуществляется после очередного выполнения тела цикла (в конце итерации). Если обозначить параметр цикла через i , начальное значение через i_n , конечное i_k , а изменение параметра $i = i + i_{ш}$, то оператор можно записать так:

<div style="display: inline-block; text-align: left;"> <u>для $i = i_n, i \leq i_k, i = i + i_{sh}$</u> <u>повторить</u> оператор </div>	или	<div style="display: inline-block; text-align: left;"> <u>для $i = i_n, i_k, i_{sh}$</u> <u>повторить</u> оператор </div>
---	-----	--

Последнюю форму записи оператора цикла с параметром используют в случаях, когда заданы начальное и конечное значения параметра цикла, а шаг изменения его постоянный. Графическая схема оператора цикла с параметром определяется конструкциями, приведенными на рисунке 2.2.

На языке C++ этот оператор имеет следующий вид:

```
for (инициализация; условие; изменение)
    оператор
```

или в общем виде:

```
for (выражение1; выражение2; выражение3)
    оператор
```

Оператор цикла с параметром, соответствующий частному случаю, выглядит так:

```
for (i=iн; i<= iк; i = i + iш )
    оператор
```

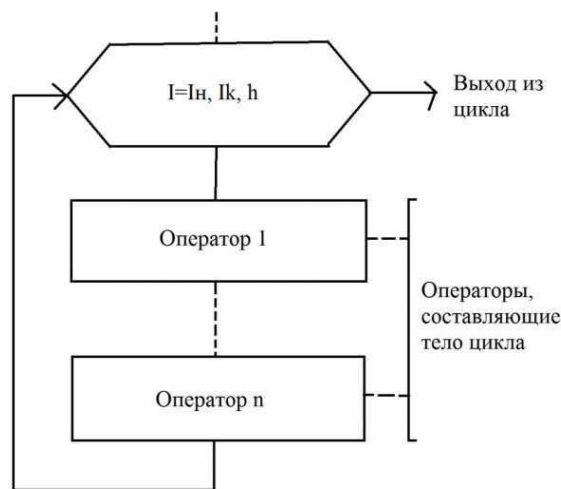


Рис. 2.2. Конструкция оператора цикла с параметром:

Пример 1. Вывести на экран значения кубов натуральных чисел от 1 до 10.

Программа с использованием цикла *while*.

```
int i; // описание переменной, отвечающей за число
i = 1; // инициализация
while (i <= 10) // проверка условия
{
    // вывод на экран значения числа и его куба:
    cout<< i << " " <<pow(i, 3) <<endl;
    i++; // увеличение значения числа
}
```

Программа с использованием цикла *for*.

```
int i; // описание переменной-счетчика
// задание начального и
//конечного значений счетчика и шага изменения:
for (i=1; i <= 10; i++) {
//вывод на экран значения числа и его куба:
    cout<< i << " " <<pow(i, 3) <<endl;
}
```

Результат работы обеих программ идентичен, однако код с использованием цикла *for* является более компактным и понятным для восприятия.

Пример 2. Написать программу вычисления факториала f натурального числа n по формуле $f = n! = 1 \cdot 2 \cdot \dots \cdot (n - 1) \cdot n$, где n вводится с клавиатуры.

Псевдокод алгоритма

Алгоритм факториал
начало

скаляры n, f, k - целые

ввод (n)

$f=1$

для $k=1, n, 1$

повторить

$f=f*k$

вывод (f)

конец

Программа на языке C++

```
#include<iostream.h>
```

```
void main (    )
```

```
{
```

```
    int k, n;
```

```
    double f;
```

```
    cout<< "Input number: ";
```

```
    cin>> n;
```

```
    f=1;
```

```
    for (k=1; k<=n; k++)
```

```
        f=f*k;
```

```
    cout<< "факториал =" << f;
```

```
}
```

Пример 3.С клавиатуры вводятся целые числа. Суммировать числа, пока пользователь подтверждает продолжение ввода. Считать подтверждением продолжения ввода символ 'y' (yes).

Программа на языке C++.

```
// описание переменных и инициализация значения
//суммы
int number, summ=0;
// символьная переменная для подтверждения
//продолжения ввода
char ch;
do // начало цикла с постусловием
{
    cout<< "Input number: "; // запрос числа
    cin>> number; // ввод числа
    summ += number; // операция сложения
    cout<< "Continue? "; // запрос на подтверждение
                        //продолжения ввода
    cin>>ch; // считывание символа
} while (ch == 'y'); // проверка условия
cout<< "Summis " <<summ<<endl; // вывод результата
```

Пример 4. На интервале $[a, b]$ с шагом dx вычислить значения кусочно-заданной функции $F(x)$.

$$F(x) = \begin{cases} \frac{6-x}{5x^2}, & x < -3, \\ \sin 2x, & -3 \leq x \leq 3, \\ 3, & x > 3. \end{cases}$$

Программа на языке C++.

```
#include<iostream>
#include<iomanip> // Библиотека манипуляторов
#include<math.h> // Математическая библиотека
int main()
{
    double a, b, x, dx, f;
    cout<<left; // выравнивание по левому краю
    cout<< "Input a: "; // Запрос начала интервала
    cin>>a; // Ввод начала интервала
    cout<< "Input b: "; // Запрос конца интервала
```

```

cin>>b;                // Ввод конца интервала
cout<< "Input dx: "; // Запрос шага изменения
                        // аргумента
cin>>dx;                // Ввод шага изменения аргумента
// Проверка корректности ввода значений:
if ((a<= b) && (dx> 0))
{
// Вывод заголовка
// (Использование манипуляторов для выравнивания)
cout<<endl<<setw(15) << "Results" <<
                                endl<<endl;
cout<<left<<setw(15) << "x" <<setw(15)<<
    "F(x)" <<endl;

for (x = a; x <= b; x += dx)
{
    if (x < -3)
        f = (6-x) / (5*pow(x,2));
    else
        if (x<=3)
            f = sin(2*x);
        else
            f = 3;
    // Вывод результатов на экран:
    cout<<setw(15) <<x<<setw(15) <<f<<endl;
}
}
else
    cout<< "Incorrect data" <<endl;
return 0;
}

```

Для вычисления значения функции необходимо подключить заголовочный файл `<math.h>` или `<cmath.h>`, которые разработаны для выполнения математических операций. Большинство функций привлекают использование чисел с плавающей точкой. Все эти функции принимают значения `double`, если не определено иначе. Для работы с типами `float` и `longdouble` используются функции с постфиксами `f` и `l` соответственно. Все функции, принимающие или возвращающие угол, работают с радианами.

Также в программе используются манипуляторы вывода из заголовочного файла `<iomanip>`, в частности, выравнивание по левому краю `left` и установка ширины поля вывода `setw`.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Получение варианта задания на лабораторную работу.
2. Разработка алгоритма и графической схемы решения задачи.
3. Составление программы на языке C++.
4. Отладка программы.
5. Тестирование программы.
6. Составление отчета о проделанной работе.

4. ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТА

1. Оформить титульный лист с указанием темы работы.
2. Сформулировать цель и задачи работы.
3. Привести формулировку задания.
4. Построить графическую схему алгоритма.
5. Записать текст программы.
6. Привести результаты тестирования программы.
7. Сформулировать вывод по проделанной работе.

5. ВАРИАНТЫ ЗАДАНИЙ

На интервале $[a, b]$ с шагом dx вычислить значения кусочно-заданной функции $F(x)$. Значения a, b, dx вводятся пользователем. При выводе результатов провести выравнивание при помощи манипуляторов. Уметь организовать выполнение задания с помощью всех видов циклов

$$\begin{aligned}
 1. \quad F(x) &= \begin{cases} \sin 2x + tg \frac{x}{x+1}, & x \leq -2, \\ \sqrt[3]{|4x+1|} - \ln(3+x), & -2 < x \leq 1, \\ 7, & x > 1. \end{cases} \\
 2. \quad F(x) &= \begin{cases} \sqrt[4]{|x^{\sin x} - 2x - 1|^3}, & x < 1, \\ \frac{2 \arctg(x+1) + 3x^7}{\cos^2 \ln x + 1}, & 1 \leq x < 4, \\ \pi, & x \geq 4. \end{cases} \\
 3. \quad F(x) &= \begin{cases} \sqrt[3]{|25x+1|} - \cos \pi x, & x \leq 0, \\ 4tg \frac{\pi x}{5} + \sin 8x, & 0 < x < 4, \\ 11, & x \geq 4. \end{cases} \\
 4. \quad F(x) &= \begin{cases} \sin \frac{x}{6} - 3\sqrt{|x|^k}, & x < -1,5, \\ \log_2(10 + |x|) + 10 \cos \frac{2x+3}{7}, & -1,5 \leq x \leq 0, \\ 1, & x > 0. \end{cases} \\
 5. \quad F(x) &= \begin{cases} \sqrt[5]{|x^3 - 4| + 1} + 4 \sin \frac{x\pi}{5}, & x < 2, \\ 2 - e^x + tg \frac{x}{x+1}, & 2 \leq x \leq 6, \\ -4, & x > 6. \end{cases} \\
 6. \quad F(x) &= \begin{cases} 3x^2 + \sqrt[5]{|10|x|^3 + 2|}, & x \leq -3, \\ \sin \frac{\pi x}{5} - 3\sqrt{|\ln x|^{x+2} + 1}, & -3 < x \leq 1, \\ 1, & x > 1. \end{cases} \\
 7. \quad F(x) &= \begin{cases} x + 2\sqrt[3]{|5 \sin x| + 2} - \cos \pi x, & x < 2, \\ \frac{3tg \frac{\pi x}{7} + \sin \cos 4x}{2x+1}, & 2 \leq x < 4, \\ 7, & x \geq 4. \end{cases}
 \end{aligned}$$

$$\begin{aligned}
8. \quad F(x) &= \begin{cases} \log_2\left(1+\frac{|3+x|}{3-x}\right)+4\sin\frac{x}{2}, x \leq 2, \\ 8\cos\frac{\pi x}{3}-\log_2\left(10+\frac{|x|}{2}\right), 2 < x < 6, \\ \pi, x \geq 6. \end{cases} \\
9. \quad F(x) &= \begin{cases} \sqrt[3]{|24-x^2|}-\cos\frac{x-1}{x+1}, x < 5, \\ 2+tg\frac{\pi x}{2}-x(|x|+1)^{x+2}, 5 \leq x \leq 9, \\ -4, x > 9. \end{cases} \\
10. \quad F(x) &= \begin{cases} \sqrt[3]{2+\left|4-\frac{\sin x}{x+1}\right|}+\cos\pi x, x < \pi, \\ e^{x+1}+3\ln(|x^2-2x|+2)+5, \pi \leq x \leq 3\pi, \\ 1, x > \pi. \end{cases} \\
11. \quad F(x) &= \begin{cases} \lg(5+|7*x|)+3\pi x, x < 1, \\ \frac{2\arctg(x+1)+3x^7}{\cos^2 \ln x}, 1 \leq x < 5, \\ -2, x \geq 5. \end{cases} \\
12. \quad F(x) &= \begin{cases} \sqrt[4]{|x^3-4x-1|^3}, x \leq -6, \\ 4tg\frac{\pi x}{10}-3\sqrt[3]{|5x^2+x|+3}, -6 < x < -4, \\ 5, x \geq -4. \end{cases} \\
13. \quad F(x) &= \begin{cases} \sqrt[4]{|x^x-\cos x+2|^5}, x < 5, \\ \frac{3x}{x+2}tg(x+1)+2\pi\cdot\lg_{10}(35+x), 5 \leq x \leq 10, \\ 10, x > 10. \end{cases} \\
14. \quad F(x) &= \begin{cases} 20-\frac{3x}{5}\cos\frac{x}{2}+tg\frac{x}{x+1}, x < -3, \\ x+2\sqrt[3]{|5x^2-\pi|+4}, -3 \leq x \leq -1, \\ 0, x > -1. \end{cases} \\
15. \quad F(x) &= \begin{cases} 2e^{x+3}+3\sin(|x^3-x|+2) x < 2, \\ \frac{\ln(x+1)+3x^5}{\cos^2 x}, 2 \leq x < 4, \\ 1, x \geq 4. \end{cases} \\
16. \quad F(x) &= \begin{cases} \lg\left((x+5)^{x+1}-x\right)+x^2\sin x, x \leq 1, \\ \cos\frac{x+7}{x-7}+3tg\frac{x}{x+2}, 1 < x < 4, \\ 11, x \geq 4. \end{cases}
\end{aligned}$$

$$17. \quad F(x) = \begin{cases} 3xe^{3x} + \sin(x^2 + 2), & x < 3, \\ \frac{x}{x+2} \operatorname{tg}(x^2 + 1) + 2\pi \cdot \ln(35 + x), & 3 \leq x \leq 4, \\ 0, & x > 4. \end{cases}$$

$$18. \quad F(x) = \begin{cases} 2\sin(2x+3) - x \cdot \sqrt[3]{4x^2+3} + 34, & x < -4, \\ \lg(x^3 + x^2 + 3) - 5\sin x + 1, & -4 \leq x \leq 4, \\ 8, & x > 4. \end{cases}$$

$$19. \quad F(x) = \begin{cases} \pi x - \frac{x}{5} \lg(15 + \frac{|3x|}{10}) + \frac{x}{x+1}, & x < 0, \\ 5 + \frac{x}{3x+2} - x + 2\sqrt[4]{8 + (|x|+1)^3}, & 0 \leq x < 4, \\ 3, & x \geq 4. \end{cases}$$

$$20. \quad F(x) = \begin{cases} 10 \cdot \sqrt[3]{|x^3|+1} - \ln(3+x), & x \leq 4, \\ \cos \frac{\pi x}{5} + \sin 3x + e^{2x}, & 4 < x < 7, \\ -3, & x \geq 7. \end{cases}$$

$$21. \quad F(x) = \begin{cases} 7\sin^3 x + \sqrt[5]{10x^2+3}, & x < -6, \\ \lg\left(1 + \frac{|3+x|}{3-x}\right) + \cos \frac{x}{2}, & -6 \leq x \leq -1, \\ -\pi, & x > 1. \end{cases}$$

$$22. \quad F(x) = \begin{cases} \sqrt[5]{|x^3-4|+1} + 4\sin \frac{x\pi}{5}, & x < 0, \\ \operatorname{ctg}(k+1) + \pi x^5, & 0 \leq x \leq 3, \\ 9, & x > 3. \end{cases}$$

$$23. \quad F(x) = \begin{cases} \log_4(|x|+x^2+5) - 4\sin x, & x < 7, \\ \sqrt[5]{|2+x^3|} - \sin \frac{2x}{4x+1}, & 7 \leq x < 9, \\ 2, & x \geq 9. \end{cases}$$

$$24. \quad F(x) = \begin{cases} \sqrt[3]{|x-4x^2|+1} + 2\sin \frac{\pi x}{3}, & x \leq 2, \\ 3\cos \frac{x+3}{2} + \operatorname{tg} \frac{x-2}{x+2}, & 2 < x < 4, \\ 11, & x \geq 4. \end{cases}$$

$$25. \quad F(x) = \begin{cases} \frac{2x}{x-5} \sin(2x+1) + 2\pi \cdot \ln(x-6), & x < -6, \\ \sqrt[3]{|4x^2|+5} - \ln\left(\frac{x+6}{x-6}\right), & -6 \leq x \leq -1, \\ 1, & x > -1. \end{cases}$$

6. КРИТЕРИИ РЕЗУЛЬТАТИВНОСТИ ВЫПОЛНЕНИЯ РАБОТЫ

Лабораторная работа считается успешно выполненной, если:

- представленный отчет содержит все необходимые пункты, согласно требованиям;
- успешно демонстрируется работа программы как на выбранных студентом исходных данных, так и на заданных преподавателем исходных данных;
- студент правильно отвечает на контрольные вопросы.

7. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое циклический алгоритм?
2. Какие виды циклов существуют, в чем их принципиальные отличия?
3. Что означает цикл с предусловием и какие операторы реализуют его в C++?
4. Что означает цикл с постусловием и как он реализуется на языке C++?
5. Как изображаются циклы в графической схеме алгоритма?
6. Что такое тело цикла и каким образом в него можно включить несколько операторов?
7. Что означает вычислить с точностью ε сумму бесконечного числа слагаемых?
8. Как записать вычисление суммы с заданной точностью ε при использовании оператора цикла `while`?
9. Назовите особенности использования цикла со счетчиком.
10. Какие функции содержит заголовочный файл `<math.h>`?
11. Какие параметры вывода можно изменить с помощью манипуляторов?