

Лабораторная работа 3

Операторы цикла в языке Си

Цель работы: изучить особенности использования операторов цикла `while`, `for` и `do while`.

Теоретический материал

Часто при создании программ на ЭВМ требуется много раз выполнить одну и ту же группу операторов. Например, для вычисления суммы ряда длиной N или перебора элементов массива с целью определения наибольшего или наименьшего значения и т.п. Во всех этих случаях необходим инструмент для реализации повторяющихся операций и таким инструментом являются операторы цикла.

Оператор цикла `while`

С помощью данного оператора реализуется цикл, который выполняется до тех пор, пока истинно условие цикла. Синтаксис данного оператора следующий:

```
while(<условие>)  
{  
    <тело цикла>  
}
```

Приведем пример реализации данного цикла, в котором выполняется суммирование элементов ряда $s = \sum_{i=0}^{\infty} i$, пока $S < N$:

```
int N=20, i = 0;  
long S = 0L;  
while(S < N)  
{  
    S=S+i;  
    i++;  
}
```

В данном примере реализуется цикл *while* с условием $i < N$. Так как начальное значение переменной $i=0$, а $N=20$, то условие истинно и выполняется тело цикла, в котором осуществляется суммирование переменной i и увеличение ее на 1. Очевидно, что на 20 итерации значение $i=20$, условие станет ложным и цикл будет завершен. Продемонстрируем гибкость языка С, изменив данный пример следующим образом:

```
int N=20, i = 0;  
long S = 0L;  
while((S=S+i++) < N);
```

В данном случае при проверке условия сначала выполняются операторы, стоящие в скобках, где и осуществляется суммирование элементов ряда и только, затем, проверяется условие. Результат выполнения обоих вариантов программ одинаковый и $S=21$. Однако последняя конструкция бывает удобной при реализации опроса клавиатуры, например, с помощью функции *scanf()*:

```
int num;
while (scanf("%d", &mun) == 1)
{
    printf("Вы ввели значение %d\n", num);
}
```

Данный цикл будет работать, пока пользователь вводит целочисленные значения и останавливается, если введена буква или вещественное число. Следует отметить, что цикл *while* можно принудительно завершить даже при истинном условии цикла. Это достигается путем использования оператора *break*. Перепишем предыдущий пример так, чтобы цикл завершался, если пользователь введет число 0.

```
int num;
while (scanf("%d", &mun) == 1)
{
    if (num == 0) break;
    printf("Вы ввели значение %d\n", num);
}
```

Цикл завершается сразу после использования оператора *break*, т.е. в приведенном примере, при вводе с клавиатуры нуля функция *printf()* выполняться не будет и программа перейдет на следующий оператор после *while*. Того же результата можно добиться, если использовать составное условие в цикле:

```
int num;
while (scanf("%d", &mun) == 1 && num != 0)
{
    printf("Вы ввели значение %d\n", num);
}
```

Таким образом, в качестве условия возможны такие же конструкции, что и в операторе *if*.

Оператор цикла *for*

Работа оператора цикла *for* подобна оператору *while* с той лишь разницей, что оператор *for* подразумевает изменение значения некоторой переменной и проверки ее на истинность. Работа данного оператора продолжается до тех пор, пока истинно условие цикла. Синтаксис оператора *for* следующий:

```
for (<инициализация счетчика>; <условие>; <изменение значения>
```

```

счетчика>)
{
<тело цикла>
}

```

Рассмотрим особенность реализации данного оператора на примере вывода таблицы кодов ASCII символов.

```

char ch;
for(ch = 'a'; ch <= 'z'; ch++)
printf("Значение ASCII для %c - %d.\n",ch,ch);

```

В данном примере в качестве счетчика цикла выступает переменная *ch*, которая инициализируется символом 'a'. Это означает, что в переменную *ch* заносится число 97 – код символа 'a'. Именно так символы представляются в памяти компьютера. Код символа 'z' – 122, и все малые буквы латинского алфавита имеют коды в диапазоне [97; 122]. Поэтому, увеличивая значение *ch* на единицу, получаем код следующей буквы, которая выводится с помощью функции *printf()*. Учитывая все вышесказанное, этот же пример можно записать следующим образом:

```

for(char ch = 97; ch <= 122; ch++)
    printf("Значение ASCII для %c - %d.\n",ch,ch);

```

Здесь следует отметить, что переменная *ch* объявлена внутри оператора *for*. Это особенность языка C - возможность объявлять переменные в любом месте программы.

Существует много особенностей реализации данного оператора, отметим основные из них, которые могут заметно повысить скорость написания программ. Следующим примером продемонстрируем особенности изменения значения счетчика цикла.

```

int line_cnt = 1;
double debet;
for(debet = 100.0; debet < 150.0; debet = debet*1.1,line_cnt++)
    printf("%d. Ваш долг теперь равен %.2f.\n",line_cnt, debet);

```

Следующий фрагмент программы демонстрирует возможность программирования сложного условия внутри цикла.

```

int exit = 1;
for(int num = 0; num < 100 && !exit; num += 1)
{
    scanf("%d",&mov);
    if(mov == 0)
        exit = 0;
    printf("Произведение num*mov = %d.\n",num*mov);
}

```

Оператор *for* с одним условием:

```
int i=0;
for(;i < 100;) i++;
```

и без условия

```
int i=0;
for(;;) {i++; if(i > 100) break;}
```

В последнем примере оператор *break* служит для выхода из цикла *for*, т.к. он будет работать «вечно» не имея никаких условий.

Оператор цикла **do while**

Все представленные выше операторы циклов, так или иначе, проверяют условие перед выполнением цикла, благодаря чему существует вероятность, что операторы внутри цикла никогда не будут выполнены. Такие циклы называют циклы с предусловием. Однако бывают ситуации, когда целесообразно выполнять проверку условия после того, как будут выполнены операторы, стоящие внутри цикла. Это достигается путем использования операторов *do while*, которые реализуют цикл с постусловием. Следующий пример демонстрирует реализацию такого цикла.

```
const int secret_code = 13;
int code_ent;
do
{
    printf("Введите секретный код: ");
    scanf("%d",&code_ent);
}
while(code_ent != secret_code);
```

Из приведенного примера видно, что цикл с постусловием работает до тех пор, пока истинно условие, т.е. в данном случае пока значение введенного кода будет отличаться от значения секретного кода. Также следует обратить внимание на то, что после ключевого слова *while* должна стоять точка с запятой. При реализации данного цикла можно использовать составные условия, подобно циклу *while*, а также принудительно выходить из цикла с помощью оператора *break*.

Задание на лабораторную работу

Использование условного оператора совместно с оператором цикла

Решить три задания с использованием циклов и условного оператора. Для каждой программы составить блок-схему алгоритма.

1. Использование условного оператора в теле операторов цикла с условием

1. Дано натуральное число. Определить:
 - а) количество цифр 3 в нем;
 - б) сколько раз в нем встречается последняя цифра;
2. Дано натуральное число. Определить:
 - а) количество четных цифр в нем. Составное условие и более одного неполного условного оператора не использовать;
 - б) сумму его цифр, больших пяти;
3. Дано натуральное число. Определить:
 - а) произведение его цифр, больших семи;
 - б) сколько раз в нем встречаются цифры 0 и 5 (всего).
4. Дано натуральное число. Определить:
 - а) сколько раз в нем встречается цифра a ;
 - б) количество его цифр, кратных z (значение z вводится с клавиатуры)
5. Дано натуральное число. Определить:
 - а) сумму его цифр, больших a (значение a вводится с клавиатуры; $0 < a < 8$);
 - б) сколько раз в нем встречаются цифры x и y .
6. Дана непустая последовательность целых чисел (организовать ввод значений в цикле), оканчивающаяся нулем. Найти:
 - а) сумму всех чисел последовательности, больших числа x ;
 - б) количество всех четных чисел последовательности.

7. Дана последовательность ненулевых целых чисел (организовать ввод значений в цикле), оканчивающая нулем. Определить, сколько раз в этой последовательности меняется знак. (Например, в последовательности 10, -4, 12, 56, -4 знак меняется 3 раза.)

8. Дано натуральное число. Определить его максимальную цифру.

9. Дано натуральное число. Определить его минимальную цифру.

10. Дано натуральное число. Определить номер цифры 8 в нем, считая от конца числа. Если такой цифры нет, ответом должно быть число 0, если таких цифр в числе несколько — должен быть определен номер самой левой из них.

11. Напечатать все кратные тринадцати натуральные числа, меньшие 100.

12. Найти 15 первых натуральных чисел, делящихся нацело на 19 и находящихся в интервале, левая граница которого равна 100.

13. Найти 20 первых натуральных чисел, делящихся нацело на 13 или на 17 и находящихся в интервале, левая граница которого равна 500.

14. Найти 10 первых натуральных чисел, оканчивающихся на цифру 7, кратных числу 9 и находящихся в интервале, левая граница которого равна 100.

2. Использование условного оператора после операторов цикла с условием

1. Дано натуральное число. Верно ли, что сумма его цифр больше 10?

2. Дано натуральное число. Верно ли, что произведение его цифр меньше 50?

3. Дано натуральное число. Верно ли, что количество его цифр есть четное число?

4. Дано натуральное число. Верно ли, что это число четырехзначное?

5. Дано натуральное число.

а) Верно ли, что его первая цифра не превышает 6?

- б) Верно ли, что оно начинается и заканчивается одной и той же цифрой?
6. Дано натуральное число. Определить, какая из его цифр больше: первая или последняя.
7. Дано натуральное число. Верно ли, что сумма его цифр меньше a ?
8. Дано натуральное число. Верно ли, что произведение его цифр больше b ?
9. Дано натуральное число.
- а) Верно ли, что это число k -значное? Составное условие и вложенный условный оператор не использовать.
- б) Верно ли, что его первая цифра превышает m ?
10. Дано натуральное число.
- а) Верно ли, что сумма его цифр больше k , а само число четное?
- б) Верно ли, что количество его цифр есть четное число, а само число не превышает b ?
11. Дано натуральное число.
- а) Верно ли, что оно начинается цифрой x и заканчивается цифрой y ?
- б) Верно ли, что произведение его цифр меньше a , а само число делится на b ?
12. Дано натуральное число. Определить:
- а) Верно ли, что сумма его цифр больше m , а само число делится на n ?
- б) есть ли в нем цифры 2 и 5.
13. Дано натуральное число.
- а) Определить, есть ли в нем цифра a .
- б) Верно ли, что в нем нет цифры b ?
14. Дано натуральное число.
- в) Верно ли, что цифра a встречается в нем более k раз?
- г) Определить, есть ли в нем цифры a и b .

3. Сочетание оператора цикла и условного оператора

1. Вывести на экран все целые числа от 100 до 200, кратные трем.

2. Вывести на экран все целые числа от a до b , кратные некоторому числу c .
3. Найти сумму положительных нечетных чисел, меньших 50.
4. Найти сумму целых положительных чисел из промежутка от a до b , кратных четырем.
5. Составить программу поиска трехзначных чисел, которые при делении на 47 дают в остатке 43, а при делении на 43 дают в остатке 47.
6. Составить программу поиска четырехзначных чисел, которые при делении на 133 дают в остатке 125, а при делении на 134 дают в остатке 111.
7. Определить количество натуральных чисел из интервала от 100 до 500, сумма цифр которых равна 15.
8. Определить количество трехзначных натуральных чисел, сумма цифр которых равна целому числу n ($0 < n < 27$).
9. Найти все двузначные числа, которые делятся на n или содержат цифру n .
10. Найти сумму целых положительных чисел, больших 30 и меньших 100, кратных трем и оканчивающихся на 2, 4 и 8.
11. Дано натуральное число. Получить все его делители.
12. Дано натуральное число. Найти сумму его делителей.
13. Дано натуральное число. Найти сумму его четных делителей.
14. Дано натуральное число. Определить количество его делителей.

Задание 4: написать блок-схему и программу задачи согласно условию.

№	Задание
1	Даны два целых числа A и B ($A < B$). Найти все целые числа, расположенные между данными числами (включая сами эти числа), в порядке их возрастания, а также количество N этих чисел.
2	Даны два целых числа A и B ($A < B$). Найти все целые числа, расположенные между данными числами (не включая сами эти числа), в порядке их убывания, а также количество N этих чисел.
3	Дано вещественное число A и целое число N (> 0). Найти A в степени N : $AN = A \cdot A \cdot \dots \cdot A$ (числа A перемножаются N раз) .
4	Дано вещественное число A и целое число N (> 0). Найти все целые степени числа A от 1 до N .
5	Дано вещественное число A и целое число N (> 0). Вывести $1 + A + A^2 + A^3 + \dots + A^N$. Дано вещественное число A и целое число N (> 0). Найти $1 - A + A^2 - A^3 + \dots + (-1)^N A^N$.
6	Дана последовательность чисел от 1 до N и число M (> 1). Найти наименьшее целое среди последовательности чисел, при котором выполняется неравенство $3K > N$, и само значение $3K$.
7	Дана последовательность чисел от 1 до N и число M (> 1). Найти наибольшее целое среди последовательности чисел, при котором выполняется неравенство $3K < N$, и само значение $3K$.
8	Дано вещественное число A (> 1). Вывести наименьшее из целых чисел N , для которых сумма $1 + 1/2 + \dots + 1/N$ будет больше A , и саму эту сумму .
9	Дано целое число N (> 0). Найти произведение $1 \cdot 2 \cdot \dots \cdot N$. Чтобы избежать целочисленного переполнения, вычислять это произведение с помощью вещественной переменной и выводить его как вещественное число.
10	Дано целое число N (> 0). Найти произведение $2 \cdot 1/(2) \cdot 1/(3) \cdot \dots \cdot 1/(N)$.
11	Дано вещественное число X и целое число N (> 0). Найти значение $1 + X + X^2/2 + \dots + X^N/N$.
12	Дано вещественное число X и целое число N (> 0) . Найти значение $X - X^3/3 + X^5/5 - \dots + (-1)^N X^{2N+1}/(2N+1)$.
13	Дано вещественное число X и целое число N (> 0). Найти значение $1 - X^2/2 + X^4/4 - \dots + (-1)^N X^{2N}/(2N)$.
14	Дано вещественное число X ($ X < 1$) и целое число N (> 0). Найти значение $X - X^2/2 + X^3/3 - \dots + (-1)^N X^N/N$. Полученное число является приближенным значением функции \ln в точке $1+X$.

Задание 5: составить схему алгоритма и написать программу вычисления суммы (произведения) конечного ряда. При разработке программы предусмотреть пропуск слагаемых (множителей), равных нулю или бесконечности.

№ вар.	Задание	№ вар.	Задание
1	$W = \sum_{i=-2}^k \frac{(-1)^i (i+3)!}{2(i-4)}$ Значение k ввести с клавиатуры	9	$Z = \prod_{n=-3}^k \frac{(n+2) n-4 }{(n+3)!}$ Значение k ввести с клавиатуры
2	$U = \prod_{t=2}^k \frac{tx^t}{t-12} \sum_{i=1}^t \left(\frac{i-4}{i-6} \right)$ Значения x и k ввести с клавиатуры	10	$W = \sum_{i=-3}^k \frac{(-1)^i}{(i-5)^2} \prod_{n=i}^{2k} \frac{n^3-8}{n+4}$ Значение k ввести с клавиатуры
3	$S = \sum_{k=1}^n \frac{(-3)^{3k+1}}{2(k-2)x^{3k+1}} \prod_{m=1}^{k+2} \frac{m^3-8}{m-4}$ Значения x и n ввести с клавиатуры	11	$P = \prod_{j=-1}^k \frac{(j-j^2)j}{j+14} \sum_{i=j}^{k+3} \frac{\sqrt[4]{ i-5 }}{ i-7 }$ Значение k ввести с клавиатуры
4	$Q = \sum_{k=1}^n \frac{(-1)^k (k-7)}{2(n-k)!}$ Значение n ввести с клавиатуры	12	$S = \sum_{k=1}^n \frac{(-1)^k (k^3-27)}{3(k+2)!}$ Значение n ввести с клавиатуры
5	$W = \sum_{i=1}^9 \frac{((7-x))^i}{(i-3)^5} \prod_{n=i}^{17} \frac{n^3-8}{n-12}$ Значение x ввести с клавиатуры	13	$Z = \prod_{j=-4}^k \frac{k^{(j+2)}j}{j-3} \sum_{i=j}^{k+5} \left(\frac{\sqrt[5]{i+5}}{i-11} - k^{5i} \right)$ Значение k ввести с клавиатуры
6	$W = \sum_{i=-2}^k \frac{(-1)^i (i+3)!}{i-4}$ Значение k ввести с клавиатуры	14	$U = \prod_{t=2}^k \frac{tx^t}{t-3} \sum_{i=1}^t \left(\frac{i-2}{i-7} \right)$ Значения x и k ввести с клавиатуры
7	$Q = \sum_{k=1}^n \frac{(-1)^k (x+3)^2}{k!}$ Значения x и n ввести с клавиатуры	15	$A = \prod_{j=1}^k \frac{(j^2-4)j}{j-k+1} \sum_{i=j}^9 \frac{i-3}{i-7}$ Значение k ввести с клавиатуры
8	$P = \prod_{j=-1}^k \frac{(j-j^2)k}{j-4} \sum_{i=j}^{k+2} \frac{\sqrt[3]{ i-5 }}{ i-7 }$ Значение k ввести с клавиатуры	16	$S = \sum_{k=1}^n \frac{(k-4)^3 (k+7)}{k!}$ Значение n ввести с клавиатуры