**Лабораторная работа 7, 8**

**Вложенные циклы**

**Цели**

* Познакомить с основными способами представления алгоритмов.
* Научиться решать задачи на составление алгоритмов вложенной циклической структуры.

**Вопросы к допуску и защите:**

1 Понятие алгоритма.

2 Способы представления алгоритмов.

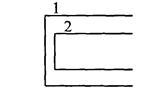
3 Свойства алгоритмов.

4 Алгоритмы циклической структуры. Вложенные циклы.

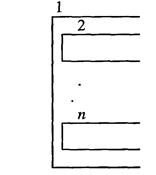
**Ход работы**

**Определение.** Цикл называется сложным (кратным, вложенным), если внутри его содержится один или несколько других циклов.

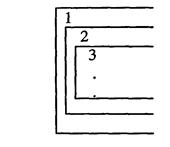
Внутренний цикл должен быть целиком внутри внешнего цикла. Сложный цикл схематично можно представить так:



внешний цикл (1) содержит один внутренний (2);

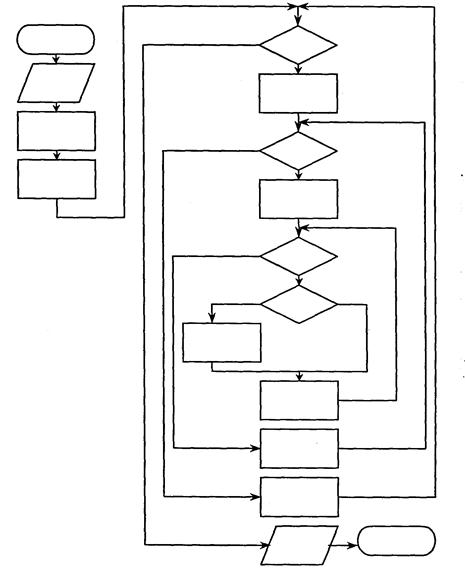


внешний цикл (1) содержит независимые внутренние циклы (2) ... (п);

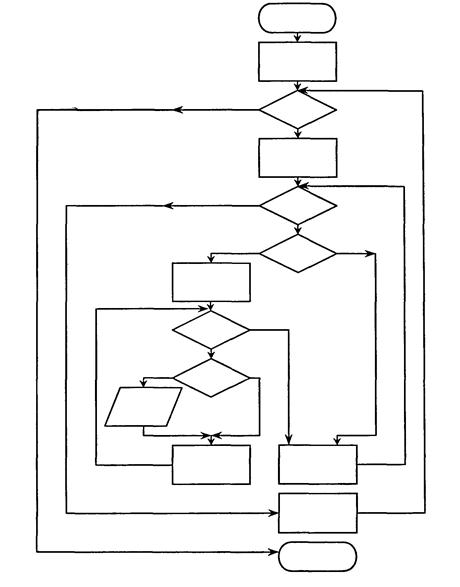


внешний цикл (1) содержит внутренний цикл (2), который содержит внутренний цикл (3) и т.д.

**Задача 1.** Заполните шаблон блок-схемы алгоритма для определения количества трехзначных натуральных чисел, сумма цифр которых равна *п.* (Операцией деления не пользоваться).



**Задача** 2. Найти все трехзначные числа, в десятичной записи которых нет одинаковых цифр. (Операцией деления не пользоваться).



**Задачи для самостоятельного решения**

1. Выведите на экран таблицу умножения используя только циклы вида repeat/until.
2. Выведите на экран таблицу умножения используя только циклы вида while.
3. Выведите на экран таблицу умножения используя один цикл while и один repeat-until .
4. Пользователь вводит числа до тех пор пока не введёт число меньшее 1. В ответ на каждое введённое им число выводите на экран все нечетные числа от 1 до это числа, при этом делящиеся на 5. Если же пользователь ввел число меньшее 1, то завершите программу.
5. Пользователь вводит первое целое число-ограничитель *m*. А затем начинает вводить целые числа по одному, пока не введёт число большее числа-ограничителя.  
   Если очередное целое число больше 1, то в ответ на каждое такое число программа должна выводить все целые числа от единицы до этого числа.

**Примечание:** это задача на вложенные циклы, в качестве внешнего надо использовать while, а в качестве внутреннего можно использовать или for или while.

1. Пользователь вводит целое положительное число, если оно не соответствует критериям (то есть не является целым и положительным), выведете сообщение об ошибке, в противном случае выведете на экран все числа от 1 до введённого пользователем.
2. Модифицируйте предыдущую задачу так, чтобы в случае, если число удовлетворяет требованиям (целое, положительное), то на экран выводились четные числа.
3. Пользователь вводит числа до тех пор пока им не будет передан ноль. В ответ на каждое число программа должна сообщать чётное оно или нет.
4. С помощью вложенных циклов выведите на экран таблицу деления чисел от 1 до 9.

**Контрольные вопросы:**

1. Какое значение примет переменная *x* после выполнения программ:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| а) | |  | | --- | | x:=0;   for i:=1 to 10 do     for k:=i+1 to 10 do       x:=x+1;  **45** | | б) | |  | | --- | | x:=0;   for i:=1 to 10 do     for k:=i+1 to 10-i do       x:=x+1;  **20** | |
| в) | |  | | --- | | x:=0;   for i:=1 to 5 do     for k:=i-1 to i+1 do       x:=x+k; | | г) | |  | | --- | | x:=10;   for i:=1 to 5 do     for k:=0 to i do       x:=x+(k-i); | |

2. Что выведут программы?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| а) | |  | | --- | | for i:=1 to 4 do   begin     if i mod 2 = 0 then       n:=i+1     else       n:=i;     for k:=1 to n do       write(n-i, '   ');     writeln;   end; | | б) | |  | | --- | | for i:=1 to 3 do     for k:=3 downto 1 do       for n:=i-k to (i+k) div 2 do         write(n, '   '); | |