

**Уважаемые студенты групп!**

**Вашему вниманию представлена лекция на тему  
«ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИЙ АЛГОРИТМ В СРЕДЕ  
ПРОГРАММИРОВАНИЯ TURBO PASCAL». Лекция рассчитана на 4  
часа**

### **Задание**

1. Прочитать внимательно лекцию.
2. Законспектировать лекцию в рабочую тетрадь не менее 3-5 страницы рукописного текста. В конспекте лекции обязательно должно быть приведены примеры.
3. Решить приведенные в лекции в контрольных вопросах задачи.
4. Дата предоставления полного фотоотчета лекции будет сообщена дополнительно.

С уважением Ганзенко Ирина Владимировна

!!! Если возникнут вопросы обращаться по телефону 0721134803 (вацап),  
+79591134803 (телеграмм)

[disobuch.ganzenko2020@mail.ru](mailto:disobuch.ganzenko2020@mail.ru)

## **ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИЙ АЛГОРИТМ В СРЕДЕ ПРОГРАММИРОВАНИЯ TURBO PASCAL**

**Цель:** получение навыков программирования и решения задач с циклическим вычислительным процессом в среде программирования Turbo Pascal. Получение практических навыков работы с ЭВМ.

### **План**

- 1 Понятие цикла**
- 2 Команда цикла с параметром (for)**
- 3 Команда цикла с предусловием (while)**
- 4 Оператор цикла с постусловием (repeat-until)**
- 5 Контрольные вопросы**

### **1 Понятие цикла**

**Цикл** - это процесс выполнения определенного набора команд многократное количество раз. Если вычислительный процесс содержит многократные вычисления по одним и тем же математическими зависимостями, но для различных значений начальных величин (переменных), его называют циклическим. Величины, которые изменяются в цикле называют переменными цикла.

Алгоритм циклической структуры в общем виде должен содержать:

- 1) подготовку цикла: задания начальных значений переменных цикла перед первым его выполнением;
- 2) тело цикла: действия, повторяющиеся в цикле для различных значений переменных цикла;
- 3) модификацию: изменение значений переменных цикла перед каждым новым его повторением;
- 4) управление циклом: проверка условий продолжения (или окончания) цикла и переход к началу тела цикла, если выполняется условие продолжения цикла (или выход из цикла после его окончания).

Циклы разделяют на циклы с заданным количеством повторений и циклы с заранее неизвестным количеством повторений. Реализуют циклы или с помощью конструкции if-goto, или значительно эффективнее, с помощью команд цикла. Есть три вида команд, с помощью которых можно запрограммировать повторяющиеся фрагменты программы: с параметром, с предусловием и с постусловием.

## **2 Команда цикла с параметром (for)**

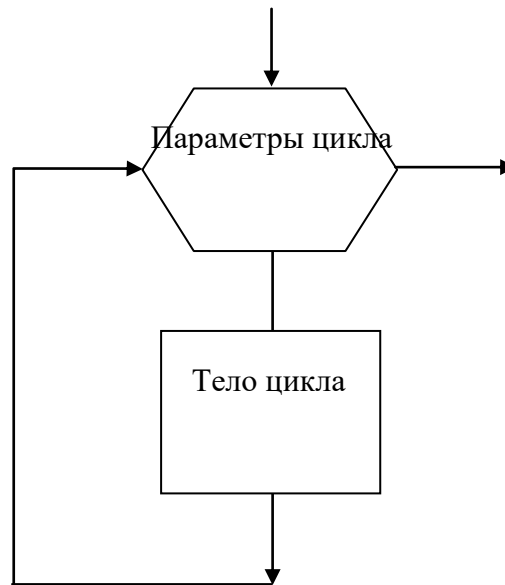
**Циклы с заданным количеством повторений** называют также циклами со счетчиком. Число повторений тела цикла в этом случае считается с помощью специальной переменной (счетчика), для которой известны начальные и конечные (пороговые) значения, а также шаг изменения. Управление циклом выполняется посредством сравнения текущих значений счетчика с заданным порогом. Переменную счетчика часто называют параметром цикла, а сам цикл - циклом с параметром.

Оператор цикла с параметром (for) имеет два вида. Рассмотрим первый:  
**for <параметр> = <выражение 1> to <выражение 2> do <оператор 1>;**

Здесь for (для), to (до), do (выполнить) - зарезервированы (служебные) слова; параметр - это переменная целого, символьного, логического или перечисленного типа, кроме действительного, а выражения 1 и 2 - это некоторые константы такого же типа, что и параметр цикла которые задают начальное и конечное значение параметра.

Действие команды. Параметру цикла присваивается значение выражения 1. Если это значение меньше или равно, чем значение выражения 2, то выполняется оператор 1. После выполнения оператору 1 значение параметра автоматически увеличивается на 1 и снова сравнивается со значением выражения 2 и т.д. Когда значение параметра станет больше, чем значение выражения 2, то выполняется следующая после цикла команда. Если во время первого сравнения окажется, что значение выражения 1 больше чем значение выражения 2, то тело цикла не выполнится ни разу.

Блок-схема цикла с параметром имеет вид:



### Примечание.

1. Указание повторения с параметром удобно использовать в тех случаях, когда предварительно можно определить количество повторений.
2. Параметр цикла должен быть описанным в разделе переменных.
3. Значение параметра цикла после окончания выполнения указания повторения обладает конечным значением, поэтому перед повторным его использованием ему необходимо вновь присвоить новое начальное значение.
4. После служебного слова `do` может стоять только один оператор. Если же в цикле необходимо выполнить группу операторов, то их запирают в операторные скобки `begin ... end`.

**Пример.** Пусть  $s = 0$  После выполнения команды цикла

`for i = 4 to 6 do begin s := s + i; z = 2 * i end;`

переменная  $s$  примет значение  $0 + 4 + 5 + 6 = 15$ , а переменная  $z = 12$ .

Замечания. Значение параметра в середине цикла изменять нельзя.

Рассмотрим второй вид команды цикла `for`:

`for <параметр> = <выражение 1> downto <выражение 2> do <оператор 1>;`

Эта команда действует как предыдущая, но значение параметра уменьшается на 1 (`downto` - вниз к).

**Пример.** Пусть  $s = 0$  После выполнения команды цикла

`for i = 6 do downto 4 begin s := s + i z = 2 * i end;`

переменная  $s$  примет значение  $0 + 6 + 5 + 4 = 15$ , а переменная  $z = 8$ .

**Задача 1.** Построить таблицу соответствия между унциями и граммами, если 1 унция = 28,353495 г. Начальное значение количества унций (uncia), шаг изменения (h) этого значения и количество строк (kil) в таблице задать самостоятельно в режиме диалога.

```

program Mira;
uses Crt;
  
```

```

const line = ' _ _ _ _ _';
var uncia, gramy, h: i, kil: integer;
begin
clrscr;
write ( 'Введите начальное значение, шаг изменения и');
writeln ( 'количество строк в таблице');
readln (uncia, h, kil)
writeln; {Формируем пустую строку}
writeln (line) {Формируем заголовок таблицы}
writeln ( 'унций грамм');
writeln (line)
for i:= 1 to kil do {Выводим таблицу на экран}
begin
gramy = 28,353495 * uncia;
writeln (uncia: 5: 2, " gramy: 10: 6)
uncia = uncif + h;
end;
writeln (line)
readln
end.

```

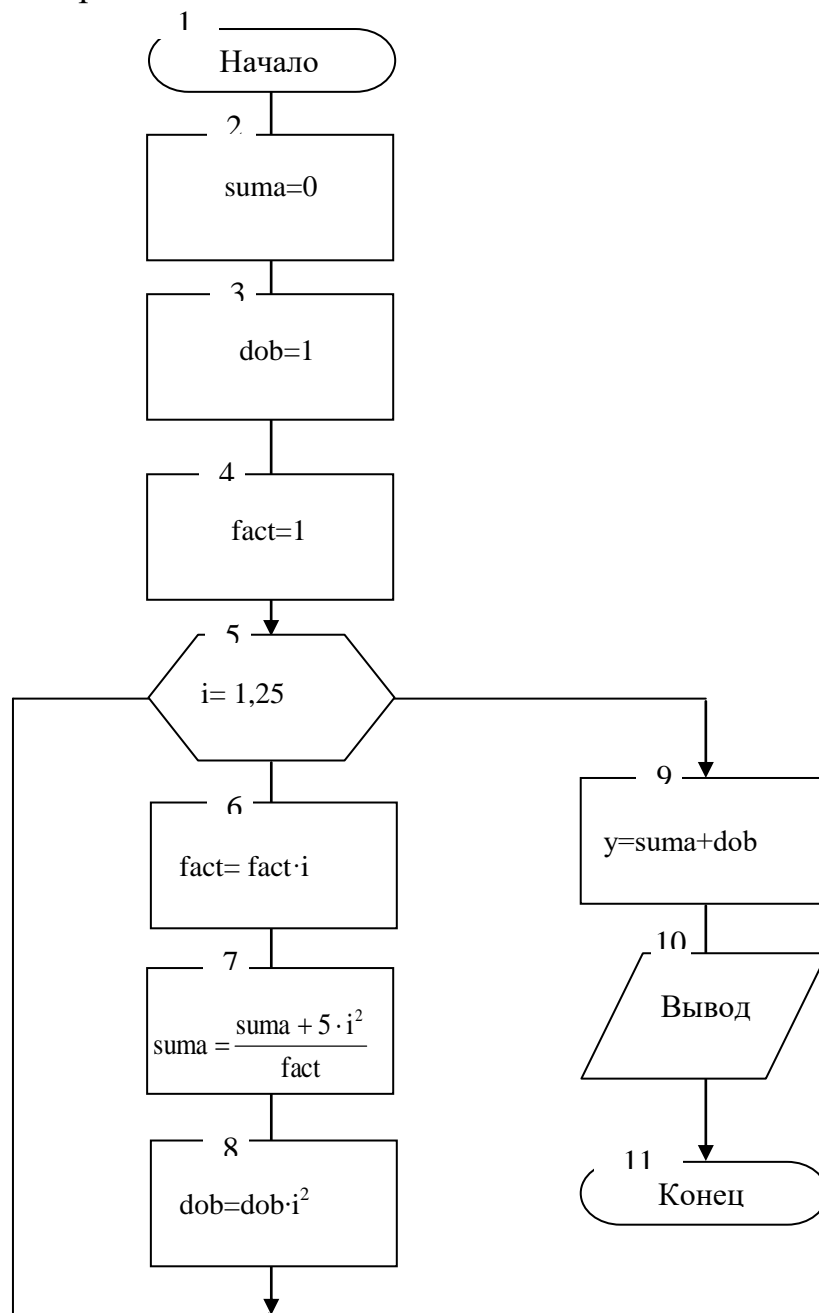
**Задача 2.** Вычислить значение выражения, содержащего сумму и произведение  $y = \sum_{i=1}^{25} \frac{5i^2}{i!} + \prod_{i=1}^{25} i^2$

```

program Suma;
uses Crt;
var i: integer;
suma, dob, y: real;
fact: longint;
begin
clrscr;
suma:= 0; {Сначала сумма равна нулю, а произведение единицы}
dob:= 1;
fact:= 1;
for i:= 1 to 25 do
begin
fact = fact * i; {Вычисляем i! = 1 * 2 * 3 * ... * i}
suma = suma + 5 * sqr (i) / fact; {Вычисляем сумму и}
dob = dob * sqr (i) {Произведение}
end;
в = suma + dob;
writeln ( 'y =', y: 7: 2)
readln
end.

```

Блок-схема алгоритма задачи 1:



### 3 Команда цикла с предусловием (while)

В большинстве задач циклы приходится выполнять неизвестное количество раз. Эта ситуация возникает, если вычисление значения функции заканчивается в случаях, когда необходимо достигнуть заданной точности, или когда выполнение оператора зависит от вводимой пользователем информации, или когда в массиве необходимо найти элемент с конкретным значением и тому подобное. В этих случаях определенную последовательность действий необходимо выполнять несколько раз, причем необходимое число повторений во время разработки программы неизвестно и может быть определен только во время работы программы, то есть в процессе вычисления задачи.

**Оператор цикла с предусловием - while (пока)** предназначен для организации многократного исполнения группы указаний (тело цикла) до тех пор, пока остается истинной условие выполнения цикла.

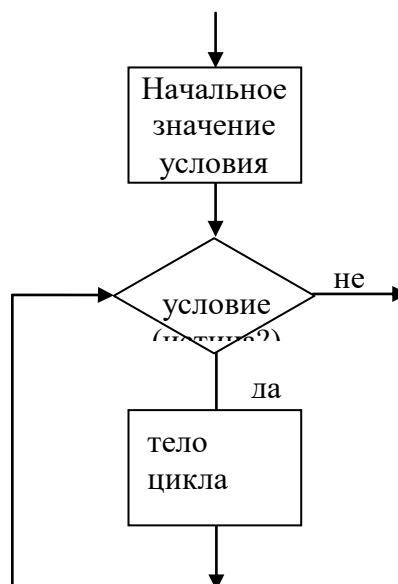
Оператор цикла с предусловием имеет вид:

```
while <логическое выражение> do  
begin  
<Указание 1>;  
<Указание 2>;  
..... ;  
<Указание N>  
end;
```

Здесь while (пока), do (выполнить) - зарезервированные слова; логическое выражение - условие, которое определяет выполнение цикла.

Действие команды. Вычисляется значение логического выражения (условия). Если условие истинно (принимает значение true), то выполняется группа указаний (тело цикла). Выполнение программы снова возвращается к проверке условия цикла. Если условие ложь (принимает значение false), то выполняется указание, которая находится после оператору повторения с предусловием.

Блок-схема оператора цикла с предусловием имеет вид:



### **Примечание.**

1. Если условие ложь при первой же проверке, то тело цикла не выполняется ни разу. Условие оператора является условием вхождения в цикл.

2. Условие оператора повторения с предусловием необходимо подбирать таким образом, чтобы в некоторый момент оно изменила свое значение true на false, иначе возникнет ситуация "зацикливания" программы.

3. Программисту необходимо самому позаботиться о необходимости увеличения (уменьшения) счетчика цикла на 1 Это можно сделать следующим образом  $I = I + 1$ .

4. Вхождение в тело оператора повторения возможно только через заголовок этого оператора.

**Пример.** Пусть переменные x, s имеют значение  $x = 4, s = 0$  После выполнения команды

```
while x <= 8 do begin s := s + x; x = x + 1 end;
```

они приобретут значений  $s = 4 + 5 + 6 + 7 + 8 = 30, x = 9$ .

Пример. Вывести таблицу чисел от 20 до 30, их квадраты и кубы с помощью команды while можно так:

```
i = 20;  
while i <= 30 do  
begin  
  writeln (i: 4, sqr (i): 6, exp (3 * ln (i)) 8);  
  i = i + 1;  
end;
```

**Задача 3.** Найти сумму всех натуральных чисел от 1 до N.

```
program SUMA1;  
var I, N, S: integer;  
begin  
  writeln ( 'Введите количество натуральных чисел:');  
  write ( 'N =');  
  readln (N)  
  I := 1;  
  S := 0;  
  while I <= N do  
  begin  
    S = S + i;  
    i = i + 1  
  end;  
  writeln ( 'S =', S)  
end.
```

**Задача 4.** протабулировать функцию  $y = \sin x$  на промежутке  $[0; 3.1]$ , с шагом  $h = 0.1$  и вычислить среднее арифметическое (sl) значений функции больших, чем 0.1 и меньших, чем 0.6 значений функции.

```
program FindSerednie;
uses Crt;
var x, y, s, sl, h, xk: real;
n: integer;
begin
  clrscr;
  x = 0; xk = 3.1; h = 0.1;
  s := 0; n := 0;
  while x <= xk do
  begin
    y = sin (x);
    writeln (x: 3: 1, y: 6: 2);
    if (y > 0.1) and (y < 0.6) then
    begin
      s = s + y;
      n = n + 1;
    end;
    x = x + h;
  end;
  if n > 0 then
  begin
    sl = s / n;
    writeln ( 'Среднее =', sl: 4: 2);
  end
  else writeln ( 'Таких значений форуме n = 0');
  readln;
end.
```

#### **4 Оператор цикла с постусловием (repeat-until)**

**Оператор цикла с постусловием предназначен** для организации многократного исполнения группы указаний (тело цикла) до тех пор, пока условие выполнения цикла не станет истинной (логическое выражение совпадает не false, а true).

Оператор цикла с постусловием имеет вид:

```
repeat
<Указание 1>;
<Указание 2>;
..... .;
<Указание N>
until <логическое выражение>;
```



Здесь repeat (повторять), until (до тех пор пока) - зарезервированные слова; логическое выражение - условие, которое определяет завершения цикла.

Проверка условия осуществляется не в начале цикла, а в конце (что гарантирует, как минимум, единовременное выполнение тела цикла).

Действие команды. Сначала выполняются указания, которые входят в тела цикла. Затем вычисляется значение условия-выражения. Если условие заблуждение (т.е. имеет значение false), то снова выполняются указания тела цикла. Если условие становится истинным (true), то выполнение цикла прекращается и выполняется следующая указание программы.

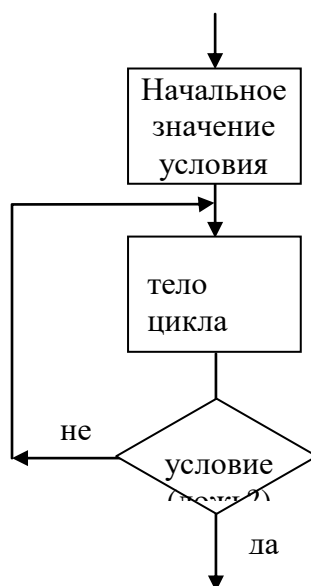
#### Примечание.

1. Тело цикла, состоящее из группы указаний, не нужно брать в скобки (begin - end), поскольку служебные слова repeat - until играют роль скобок.

2. Тело цикла в операторе цикла с постусловием выполняется минимум один раз.

3. Условие в операторе является условием выхода из цикла.

Блок-схема оператора цикла с постусловием имеет вид:



**Пример.** Пусть переменные x, y имеющих значение  $x = 5$ ,  $y = 0$ . в результате выполнения команды

repeat

$y = y + x;$

$z = 2 * x;$

$x = x - 2$

until  $x \leq 1;$

переменные y, z, x приобретут значений  $y = 0 + 5 + 3 = 8$ ,  $z = 6$ ,  $x = 1$ .

**Задача 5.** Найти сумму всех натуральных чисел от 1 до N.

```
program SUMA2;
var i, N, S: integer;
begin
  writeln ( 'Введите количество натуральных чисел:');
  write ( 'N =');
  readln (N)
  i:= 1;
  S:= 0;
  repeat
    S = S + i;
    i = i + 1;
  until i> N;
  writeln ( 'S =', S);
end.
```

**Задача 6.** протабулировать функцию  $y = \sin x$  на промежутке  $[-\pi; \pi]$ , с шагом  $h = \pi / 5$ . Результаты вычислений вывести на экран в виде таблицы. Найти максимальное (max) и минимальное (min) значения функции на этом промежутке.

Максимальное или минимальное значение функции или значение, которое удовлетворяет некоторой другому условию поиска, определяют методом сканирования (просмотра) и сравнение всех подряд значений функции с некоторым эталоном.

```
program Tabulfuncii;
uses Crt;
var x, y, max, min, h: real;
begin
  clrscr;
  x = - pi;
  h = pi / 5;
  max = sin (x) min = sin (x) {Предполагаем, что max i min}
  writeln ( 'x y'); {Находятся в первой точке}
  repeat
    y = sin (x)
    writeln (x: 5: 2, ": 3, y: 5: 2)
    if y> max then max = y; {Определяем максимум функции}
    if y <min then min = y; {Определяем минимум функции}
    x = x + h; {Увеличиваем x на h}
  until x> pi; {Проверяем условие выхода из цикла}
  writeln ( 'max =', max: 5: 2, 'min =', min: 5: 2)
  readln
end.
```

Для организации паузы в процессе выполнения программы можно использовать конструкцию Repeat Until KeyPressed (используется до нажатия любой клавиши). В языке программирования Turbo Pascal логическая функция KeyPressed контролирует нажатия клавиш. Если ни одна из клавиш не нажата, эта функция принимает значение False, иначе - True.

## 5 Контрольные вопросы

1. Что такое цикл? Приведите примеры.
2. Какой процесс называется циклическим?
3. Что такое переменная цикла. Приведите примеры.
4. Из каких составных образуется алгоритм циклической структуры?
5. Что такое тело цикла?
6. Что такое модификация цикла?
7. Как управлять циклом? Приведите примеры.
8. Что такое параметр цикла? Как он меняется?
9. Приведите пример циклического вычислительного процесса с параметром.
10. Что произойдет при использовании служебного слова to в цикле, если исходное выражение будет больше конечный?
11. Как решаются задачи с неизвестным количеством циклических повторений?
12. Приведите примеры использования оператора while.
13. В каком случае может произойти заикливание программы? Как из этого выйти?
14. В чем разница между операторами while и repeat?
15. Напишите оператор с постусловием.
16. Почему тело цикла с постусловием выполняется не реже одного раза?
17. Указать значение переменной s после выполнения оператора:
  - 1)  $s = 3; i = 1; \text{while } i < 6 \text{ do } i = i + 1; s = s + i; s = \underline{\hspace{2cm}}$
  - 2)  $s = 1; i = 0; \text{while } i < 4 \text{ do begin } i = i + 1; s = s * i; \text{end}; s = \underline{\hspace{2cm}}$
  - 3)  $s = 2; i = 1; \text{repeat } s = s + i; i = i + 1 \text{ until } i > 3; s = \underline{\hspace{2cm}}$
  - 4)  $s = 5; i = 5; \text{repeat } s = s + 1 / i; i = i - 1 \text{ until } i \leq 3; s = \underline{\hspace{2cm}}$
  - 5)  $s = 0; \text{for } i = 5 \text{ to } 7 \text{ do } s = s + 1, s = \underline{\hspace{2cm}}$
  - 6)  $s = 1; \text{for } i = 10 \text{ downto } 2 \text{ do } s = s + 1, s = \underline{\hspace{2cm}}$
18. Найти сумму целых положительных чисел кратных 7 и меньших 100.
19. Составить программу нахождения суммы цифр:  
 $S = 10 + 22 + 32 + \dots + n^2$ , если  $n = 10$ .
20. Если условие в операторе цикла while принимает значение false, то тело цикла:                     
  - а) выполняется 1 раз;
  - б) не выполняется ни разу;
  - в) наступает «заикливание».

21. Если условие в операторе цикла repeat-until принимает значение true, то тело цикла: \_\_\_\_\_

- а) выполняется 1 раз;
- б) не выполняется ни разу;
- в) наступает «зацикливание».

22. В каких операторах повторения увеличение (уменьшение) счетчика цикла на 1 предусмотрены

- 1) автоматически \_\_\_\_\_ а) while ... do
- 2) программистом \_\_\_\_\_ б) for ... to ... do
- в) repeat ... until

23. В каком случае, при  $k = 3$ , тело цикла будет выполняться:

- 1) конечное число раз \_\_\_\_\_
  - 2) ни разу \_\_\_\_\_
  - 3) наступит «зацикливание» \_\_\_\_\_
- |                        |                     |                     |
|------------------------|---------------------|---------------------|
| а) while $k \leq 5$ do | б) while $k > 5$ do | в) while $k > 3$ do |
| begin                  | begin               | begin               |
| $s = s + k;$           | $s = s + k;$        | $s = s + k;$        |
| $k = k + 1;$           | $k = k + 1;$        | $k = k + 1;$        |
| end;                   | end;                | end;                |

24. В каком случае, при  $k = 3$ , тело цикла будет выполняться:

- 1) конечное число раз \_\_\_\_\_
  - 2) ни разу \_\_\_\_\_
  - 3) наступит «зацикливание» \_\_\_\_\_
- а) for  $i = k$  to 10 do  $s = s + k;$
  - б) for  $i = k$  to 2 do  $s = s + k;$
  - в) for  $i = k$  downto 10 do  $s = s + k;$
  - г) for  $i = k$  downto 2 do  $s = s + k;$

25. В каком случае, при  $k = 3$ , тело цикла будет выполняться:

- 1) конечное число раз \_\_\_\_\_
  - 2) ни разу \_\_\_\_\_
  - 3) наступит «зацикливание» \_\_\_\_\_
- |                   |                |                |
|-------------------|----------------|----------------|
| а) repeat         | б) repeat      | в) repeat      |
| $s = s + k;$      | $s = s + k;$   | $s = s + k;$   |
| $k = k + 1;$      | $k = k + 1;$   | $k = k + 1;$   |
| until $k \leq 5;$ | until $k > 5;$ | until $k > 3;$ |

26. Указать значение переменной  $s$ , которое соответствует результату выполнения оператора:

- 1)  $s = 2; i = 0; \text{while } i < 5 \text{ do } i = i + 1; s = s + 1 / i;$  \_\_\_\_\_
  - 2)  $s = 5; i = 1; \text{while } i > 1 \text{ do begin } s = s + 1 / i; i = i - 1; \text{end};$  \_\_\_\_\_
  - 3)  $s = 1; i = 1; \text{while } i < 4 \text{ do begin } i = i + 1; s = s * i; \text{end};$  \_\_\_\_\_
  - 4)  $s = 1; i = 6; \text{while } i > 3 \text{ do begin } s = s + i; i = i - 1; \text{end};$  \_\_\_\_\_
- а) 18 б) 2.2 в) 0 г) 5 г) 10 д) 48

27. Указать значение переменной  $s$ , которое соответствует результату выполнения оператора:

- 1)  $s = 4; i = 1; \text{repeat } s = s + 1 / i; i = i - 1 \text{ until } i \leq 1;$  \_\_\_\_\_

2)  $s = 2$ ;  $i = 1$ ; repeat  $s = s + 1 / (i + 1)$ ;  $i = i + 3$  until  $i \geq 5$ , \_\_\_\_\_

3)  $s = 1$ ;  $i = 1$ ; repeat  $i = i + 1$ ;  $s = s + i$  until  $i > 1$ ; \_\_\_\_\_

4)  $s = 2$ ;  $i = 5$ ; repeat  $i = i - 1$ ;  $s = s + 1 / i$  until  $i \leq 4$ ; \_\_\_\_\_

а) 2.25 б) 2.7 в) 7 г) 10 г) 5

28. Указать значение переменной  $s$ , которое соответствует результату выполнения оператора:

1)  $s = 0$ ; for  $i = 10$  downto 6 do  $s = s + 1$ ,

2)  $s = 0$ ; for  $i = 8$  to 8 do  $s = s + 1$ ,

3)  $s = 1$ ; for  $i = 2$  downto 2 do  $s = s + 1$ ,

4)  $s = 1$ ; for  $i = 5$  to 9 do  $s = s + 1$ ,

а) 6 б) 10 в) 5 г) 2 г) 40