

## Лабораторная работа 2

### 2.1. [#10]

Определите таблицу выполнения для программы:

```
PROGRAM Copy2 (INPUT, OUTPUT);  
{Копирует первые два символа из INPUT в OUTPUT}  
VAR  
    Letter: CHAR;  
BEGIN  
    READ (Letter);  
    WRITE (Letter);  
    READ (Letter);  
    WRITELN (Letter)  
END.
```

Выполнение

```
INPUT : AZURE  
OUTPUT: AZ
```

### 2.2. Условное выполнение

#### 2.2.1. [#10]

Следующую программу разместите по строкам, сделайте отступы, отформатируйте. Оттранслируйте и определите выход (OUTPUT) для INPUT: BOB

```
PROGRAM Dup (INPUT, OUTPUT); VAR Ch1,  
Ch2, Ch3: CHAR; BEGIN READ (Ch1, Ch2,  
Ch3); IF Ch1 = Ch2 THEN WRITE (Ch1,  
Ch2, ' '); IF Ch1 = Ch3 THEN WRITE  
(Ch1, Ch3, ' '); IF Ch2 = Ch3 THEN  
WRITE (Ch2, Ch3); WRITELN END.
```

#### 2.2.2. [#10]

Следующую программу разместите по строкам, сделайте отступы, отформатируйте. Дополните эхо-входом.

Оттранслируйте и определите выход (OUTPUT) для

```
INPUT: 5  
INPUT: A5
```

```
PROGRAM What INPUT OUTPUT VAR Ch CHAR  
BEGIN READ Ch IF '0' <= Ch THEN IF Ch  
<= '9' THEN WRITELN YES ELSE WRITELN NO  
ELSE WRITELN NO END
```

## 2.3. Циклы.

### 2.3.1. [#10] (Дома)

Следующую программу разместите по строкам, сделайте отступы, отформатируйте.

Дополните аннотированным выходом.

Оттранслируйте и определите выход (OUTPUT):

```
PROGRAM Less INPUT OUTPUT VAR Ch
CHAR BEGIN READ Ch WHILE
Ch <> '#' DO BEGIN IF Ch<> ' ' THEN WRITE
Ch READ Ch END WRITELN END
```

```
INPUT:HI OUT THERE#...
```

### 2.3.2. [#20] (Дома)

Определите таблицу выполнения для программы CopyOdd.

```
INPUT:AZ#
```

```
PROGRAM CopyOdds(INPUT, OUTPUT);
{Копирует через один символы из INPUT в OUTPUT,
но до тех пор, пока не встретится первое #.}
VAR
  Ch, Next: CHAR;
  {Next - это переключатель между нечетными (odd (O))
и четными (even (E))}
BEGIN
  Next := 'O'; {odd}
  READ(Ch);
  WHILE Ch <> '#'
  DO
    BEGIN
      IF Next = 'O'
      THEN {Копирование нечетных символов}
        WRITE(Ch);
      READ(Ch);
      {Переключение величины Next}
      IF Next = 'O'
      THEN
        Next := 'E' {even}
      ELSE
        Next := 'O' {odd}
    END;
  WRITELN
END.
```

## **Правила форматирования.**

- Оператор VAR находится на одном уровне с PROGRAM. Описание переменных начинается со следующей строки. Описание переменных имеет отступ 2 символа относительно VAR. После двоеточия ставится пробел:

```
VAR
  A: CHAR;
  B, C: CHAR;
```

- Оператор присваивания (:=), а также операции сравнения и все математические операции отделяются пробелами с двух сторон:

```
A := 'b';
IF Bool = 'Y'
THEN
  ...
```

- В операторе IF ключевые слова THEN и ELSE находятся в отдельных строках на одном уровне с IF. Все, что находится внутри веток оператора, имеет отступ 2 пробела:

```
IF Bool = 'Y'
THEN
  WRITELN
ELSE
  BEGIN
    READ (Ch) ;
    WRITE (Ch)
  END;
```

- Аналогично, в операторе WHILE ключевое слово DO располагается на отдельной строке без отступа, тело цикла имеет отступ 2 пробела.