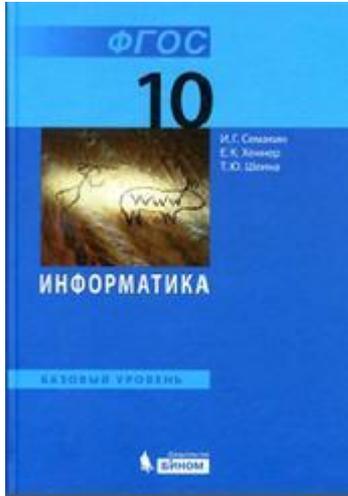


3 февраля  
Классная работа

# Выбор в языке Pascal. Оператор выбора Case

*Урок 42*

# Домашнее задание



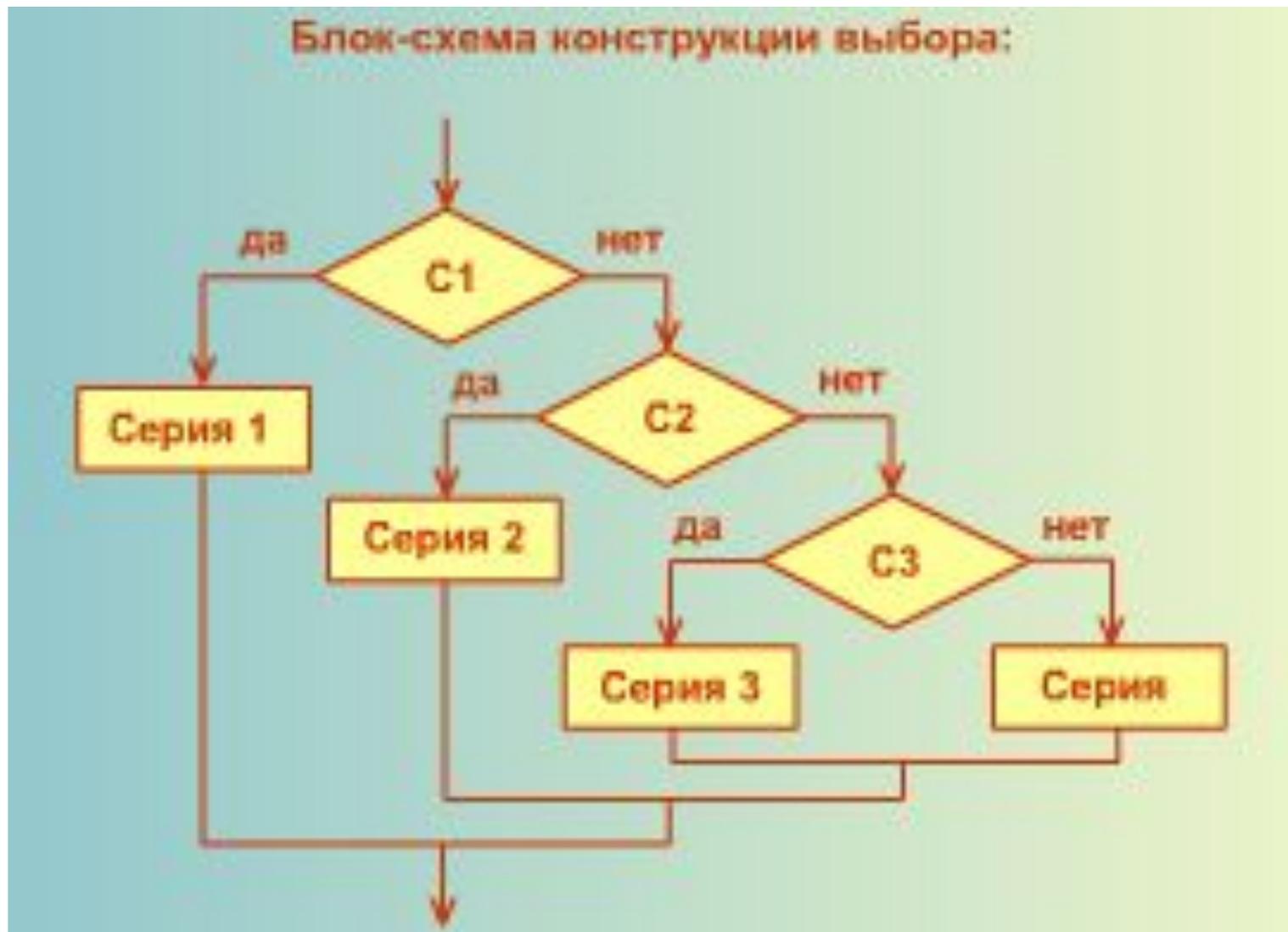
**§ 19** (стр. 132-136) – **изучить** (до конца)

**Задание 4** (стр. 136) – **на компьютере**  
в файле (отправить учителю)

**№ 2** стр. 234

**№ 5** стр. 241

Существует **еще одна** структура, в которой команды выполняются при условии, причём с несколькими ветвями — структура **выбор**:



**Пример 1.** Требуется перевести пятибалльную оценку в ее наименование: 5 — «отлично», 4 — «хорошо», 3 — «удовлетворительно», 2 — «неудовлетворительно».

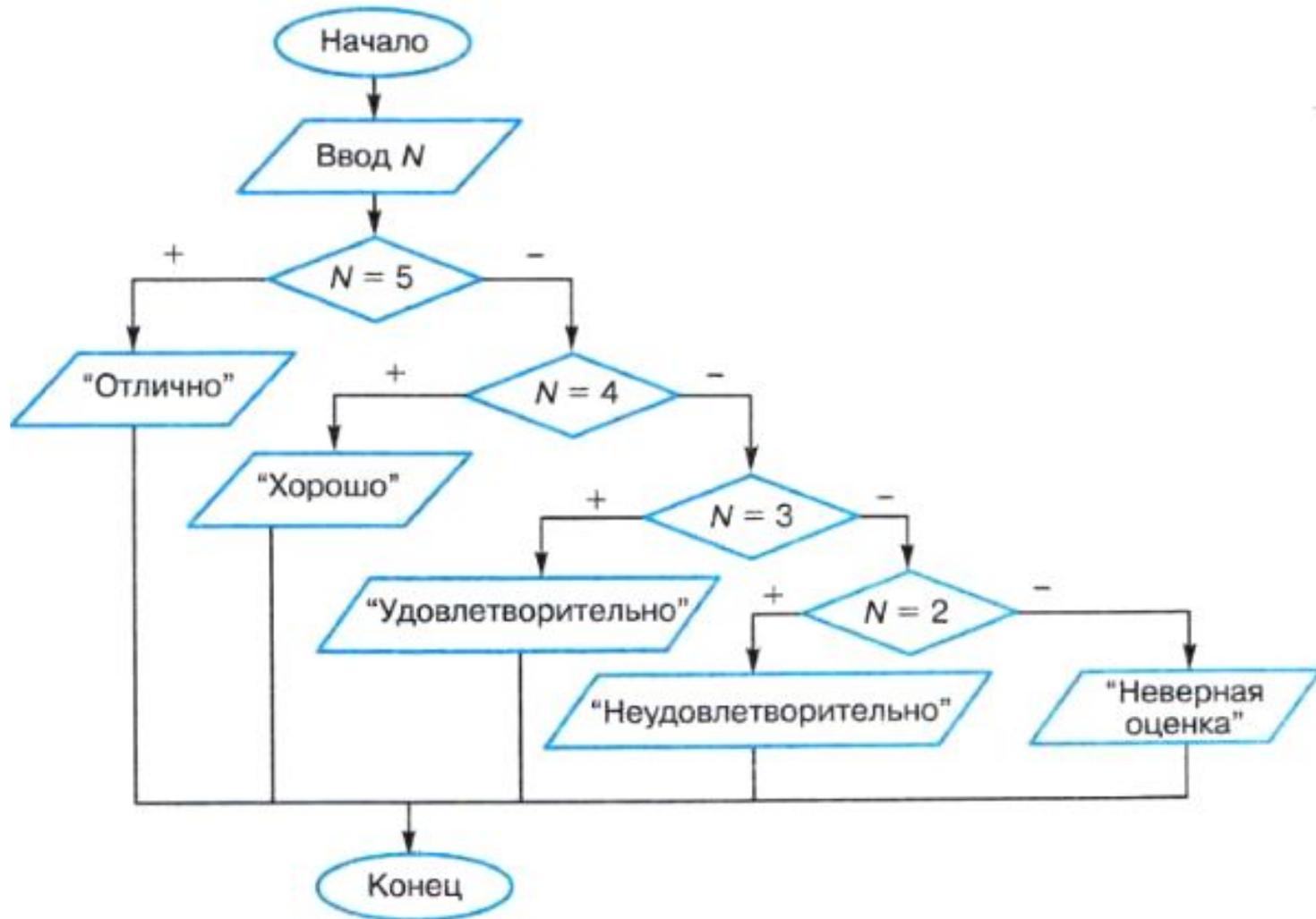


Рис. 3.14. Алгоритм перевода числовой оценки в словесную

Этот алгоритм имеет структуру вложенных ветвлений и может быть запрограммирован с использованием условного оператора **If** следующим образом:

```
Program Marks_1;
Var N: Integer;
Begin
  WriteLn('Введите оценку:');
  ReadLn(N);
  If N=5
  Then WriteLn('Отлично')
  Else If N=4
    Then WriteLn('Хорошо')
    Else If N=3
      Then WriteLn('Удовлетворительно')
      Else If N=2
        Then WriteLn('Неудовлетворительно')
        Else WriteLn('Неверная оценка')
  End.
End.
```

**Пример 2.** Решение рассмотренной в предыдущем примере задачи можно запрограммировать с помощью одного оператора выбора, имеющегося в языке Паскаль. Вот как будет выглядеть такая программа:

```
Program Marks_2;  
Var N: Integer;  
Begin  
  WriteLn('Введите оценку:');  
  ReadLn(N);  
  Case N of  
    5: WriteLn('Отлично');  
    4: WriteLn('Хорошо');  
    3: WriteLn('Удовлетворительно');  
    2: WriteLn('Неудовлетворительно');  
    Else WriteLn('Неверная оценка')  
  End;
```

# Оператор выбора

## Оператор выбора:

```
Case селектор of
  <список 1> : <действие 1>;
  <список 2> : <действие 2>;
  . . .
Else
  <действие n+1>;
End;
```

<Списки> могут быть заданы:

- 1) перечислением значений через запятую, например: 1, 4, 5;
- 2) интервалом значений, например: 1. .5, 20. .30.

Здесь **селектор** – проверяемое **целое арифметическое**, **логическое** или **символьное** выражение (выражение только какого-либо порядкового типа, не может быть вещественного типа (например real)).

# Оператор выбора

## Оператор выбора:

```
Case селектор of
  <список 1> : <действие 1>;
  <список 2> : <действие 2>;
  . . .
Else
  <действие n+1>;
End;
```

**<Списки> могут быть заданы:**

- 1) перечислением значений через запятую, например: 1, 4, 5;
- 2) интервалом значений, например: 1..5, 20..30.

**<Списки>** представляют собой перечисления возможных значений выражения селектор. При совпадении значения выражения хотя бы с одним из значений **<списка>** будет выполняться соответствующее **<действие>** и на этом выполнение команды выбора завершается.

# Примеры списков

Вариант	Пример
Одно значение	'a' 8
Перечисление значений	1, 4, 5 'a', 'b', '*', '@'
Интервалы значений	1..5, 20..30 'a' .. 'z'
Смешанные примеры	1 .. 5, 15, 55 'a' .. 'z', 'A', 'B'

**Пример 3.** В этом примере демонстрируется использование списка констант в операторе выбора. Программа сообщает, сдал студент экзамен или не сдал. Если оценка одна из следующих: 3, 4, 5, то экзамен сдан; если 2, то не сдан.

Если применить условный оператор, то эта программа запишется так:

```
If (N=3) or (N=4) or (N=5)
Then WriteLn('Экзамен сдан')
Else If N=2
    Then WriteLn('Экзамен не сдан')
    Else WriteLn('Нет такой оценки');
```

Если применить оператор выбора, то эта программа запишется так:

```
Case N of
    3, 4, 5: WriteLn('Экзамен сдан');
    2: WriteLn('Экзамен не сдан');
    Else WriteLn('Нет такой оценки')
End
```

## **ЗАДАНИЕ №4 стр. 136**

**Работу оператора множественного выбора можно проиллюстрировать следующим примером:**

**Составить программу, которая спрашивает у пользователя номер месяца и выводит соответствующее названия времени года. В случае, если пользователь укажет недопустимое число, программа должна вывести сообщение об ошибке «Ошибка ввода данных».**

**Ниже приведен рекомендованный вид экрана во время работы программы:**

**Введите номер месяца (число от 1 до 12) и нажмите <Enter> 12**

**Зима**