МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Лабораторная работа №8**

по дисциплине: Основы программирования

тема: «Использование комбинированного типа»

Выполнил: ст. группы ПВ-201

Машуров Дмитрий Русланович

Проверил:

Притчин Иван Сергеевич

Белгород 2020 г.

**Лабораторная работа №8**

**Использование комбинированного типа**

**Цель работы**: получение навыков работы с записями.

**Задания для подготовки к работе:**

1. Изучите комбинированный тип данных.
2. Разбейте задачу на подзадачи, таким образом, чтобы решение каждой подзадачи описывалось подпрограммой, а основная программа состояла бы, в основном, из вызовов подпрограмм.
3. Опишите блок-схему алгоритма укрупненных блоках.
4. Для каждой подзадачи опишите используемые структуры данных, спецификацию и блок-схему алгоритма.
5. Опишите блок-схему алгоритма решения задачи с использованием блоков «предопределенный процесс».
6. Закодируйте алгоритм.
7. Подберите наборы тестовых данных с обоснованием их выбора.

**Задания к работе:**

1. Наберите программу, отладьте ее, протестируйте.
2. Выполните анализ ошибок, выявленных при отладке программы.

**Задание варианта №17**

Известна дата и день недели некоторого события. Определить, на какой день недели приходится начало года.

**Выполнение работы:**

1. **Выделение подзадач**

Выделим следующие подзадачи:

1. Ввод даты
2. Подсчёт общего количества дней от введённой даты до 1-го января
3. Вычисление дня недели, который приходится на 1 января года введённой даты
4. **Описание структур данных**

days – массив строк с наименованием дней недели

date – комбинированный тип, описывающий дату формата *дд мм гггг день недели*

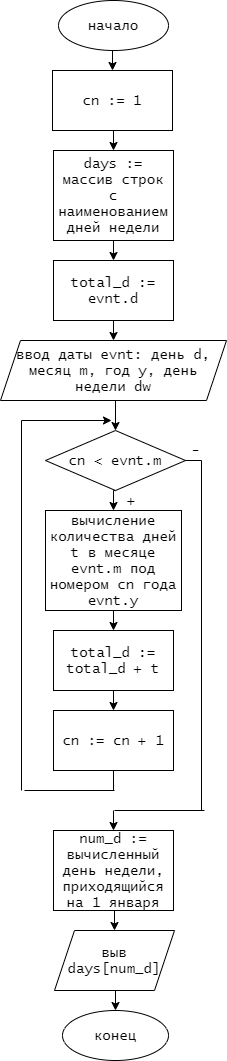
**const** days: **array**[1..7] **of** string = ('понедельник','вторник','среда','четверг','пятница','суббота','воскресенье')

**type** date = **record**

d, m, y, dw: integer;

**end**;

1. **Блок-схема в укрупнённых блоках**

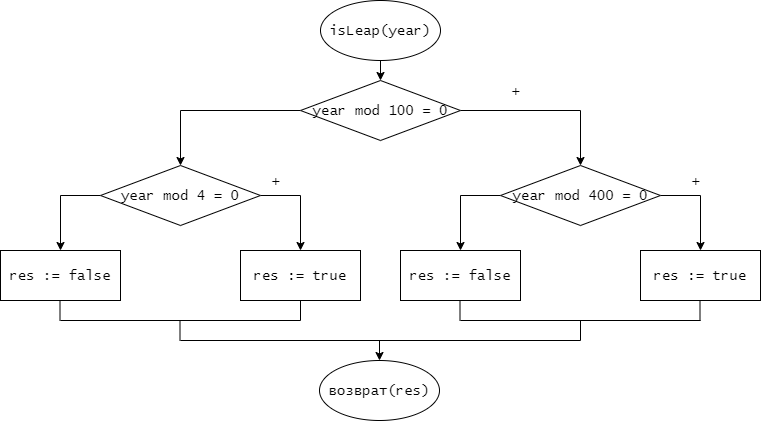
****

1. **Описание подпрограмм**

Спецификация функции isLeap

1. Заголовок: function isLeap(year: integer) : boolean
2. Назначение: возвращает значение «истина», если год year является високосным, и значение «ложь», если не является
3. Входные параметры: year
4. Выходные параметры: нет

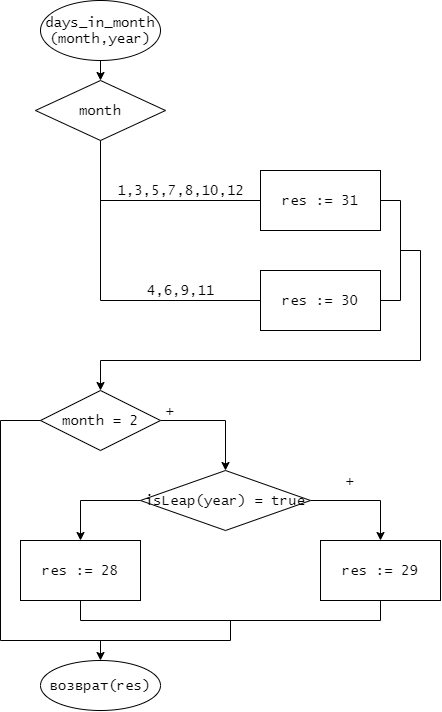
Блок-схема:



Спецификация функции days\_in\_month

1. Заголовок function days\_in\_month(month: integer; year: integer) : integer
2. Назначение: возвращает количество дней в месяце month года year
3. Входные данные: month, year
4. Выходные данные: нет

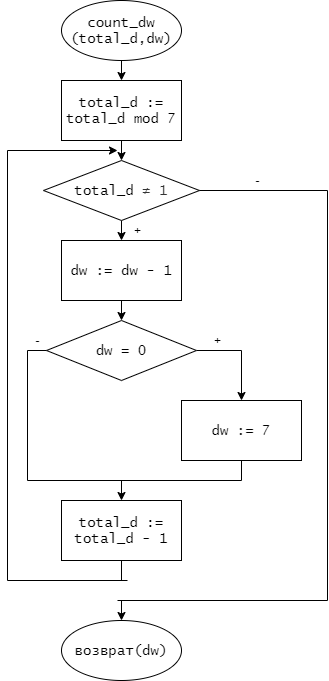
Блок-схема:



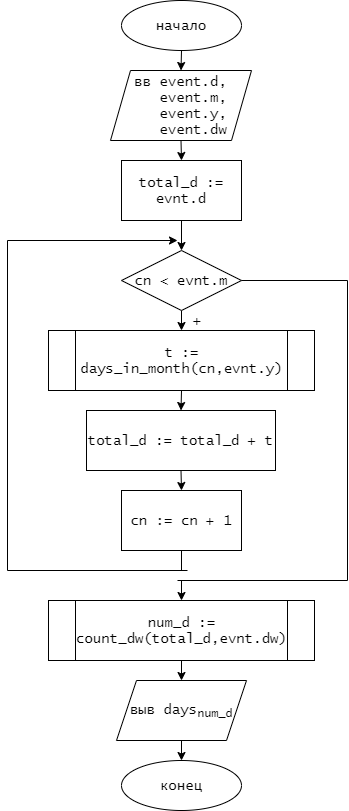
Спецификация функции count\_dw

1. Заголовок function count\_dw(total\_d: integer; dw: integer) : integer
2. Назначение: возвращает вычисленный из общего количества (от введённой даты до 1 января) дней total\_d день dw, на который приходится 1 января
3. Входные параметры: total\_d, dw
4. Выходные параметры: нет

Блок-схема:



1. **Блок-схема с блоками «предопределённый процесс»**

****

1. **Тестовые данные**

|  |  |  |
| --- | --- | --- |
| № | Дата | Результат |
| 1 | 12.09.2020, суббота | среда |
| 2 | 17.03.1994, четверг | суббота |
| 3 | 03.04.1973, вторник | понедельник |

1. **Текст программы**

**const**

days: **array**[1..7] **of** string = ('понедельник','вторник','среда','четверг','пятница','суббота','воскресенье');

**type** date = **record**

d, m, y, dw: integer;

**end**;

**function** isLeap(year: integer) : boolean;

**var** res: boolean;

**begin**

**if** (year **mod** 100 = 0) **then**

**if** (year **mod** 400 = 0) **then**

res := true

**else**

res := false

**else**

**if** (year **mod** 4 = 0) **then**

res := true

**else**

res := false;

isLeap := res;

**end**;

**function** days\_in\_month(month: integer; year: integer) : integer;

**var** res: integer;

**begin**

**case** month **of**

1, 3, 5, 7, 8, 10, 12: res := 31;

4, 6, 9, 11: res := 30;

**end**;

**if** (month = 2) **then**

**begin**

**if** (isLeap(year) = true) **then**

res := 29

**else**

res := 28;

**end**;

days\_in\_month := res;

**end**;

**function** count\_dw(total\_d: integer; dw: integer) : integer;

**begin**

total\_d := total\_d **mod** 7;

**while** (total\_d <> 1) **do**

**begin**

dw := dw - 1;

**if** (dw = 0) **then**

dw := 7;

total\_d := total\_d - 1;

**end**;

count\_dw := dw;

**end**;

**var**

evnt: date;

t, cn, total\_d, num\_d: integer;

**begin**

cn := 1;

writeln('Введите дату');

write('День: '); read(evnt.d);

write('Месяц: '); read(evnt.m);

write('Год: '); read(evnt.y);

write('День недели (число, где 1 - понедельник, 2 - вторник и т.д.): '); read(evnt.dw);

total\_d := evnt.d;

**while** cn < evnt.m **do**

**begin**

t := days\_in\_month(cn,evnt.y);

total\_d := total\_d + t;

cn := cn + 1;

**end**;

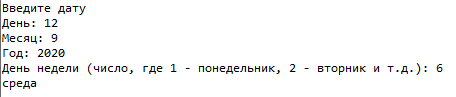
num\_d := count\_dw(total\_d,evnt.dw);

writeln(days[num\_d]);

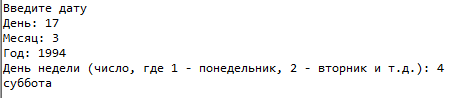
**end**.

1. **Результаты работы программы:**

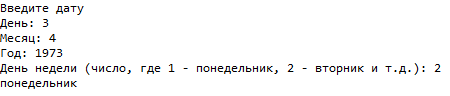
*Пример №1:*

****

*Пример №2:*

****

*Пример №3:*



1. **Анализ ошибок:**

* Отсутствие возвращаемого значения в функции days\_in\_month