Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе №6

Тема работы:

Типы пользователя

Выполнил

студент: гр. 151004 Данилов Ф.А.

Проверил: Фадеева Е.П.

Минск 2021

Содержание

[1 Постановка задачи 3](#_Toc89691675)

[2 Методика решения 4](#_Toc89691676)

[2.1 Алгоритм решения задачи 4](#_Toc89691677)

[2.2 Алгоритм нахождения дня недели 4](#_Toc89691678)

[3 Текстовый алгоритм решения задачи 5](#_Toc89691679)

[4 Структура данных 6](#_Toc89691680)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 7](#_Toc89691681)

[6 Результаты расчетов 9](#_Toc89691682)

[Приложение А 10](#_Toc89691683)

[Приложение Б 13](#_Toc89691684)

# Постановка задачи

В магазине первый понедельник каждого месяца проводится переучет товаров. Дано число, обозначающее номер года. Определить для каждого месяца этого года все числа месяцев, на которые приходится день переучета.

# Методика решения

## Алгоритм решения задачи

Для того, чтобы найти первый понедельник во всех месяцах введенного года необходимо циклом пройти через месяцы года в диапазоне 1..12.

В теле цикла нужно вложить второй цикл, который будет проходить по дням месяца, пока понедельник не будет найден. После нахождения понедельника цикл останавливается, день месяца записывается в цикл.

## Алгоритм нахождения дня недели

Найти день недели можно через формулу, предложенную Томохико Сакамото 10 марта 1993 года:

,

где:

1. – последние две цифры
2. – первые две цифры
3. – день месяца (от 1 до 31)
4. – месяц после сдвига (Март = 1, Апрель = 2, ..., Февраль = 12)
5. – день недели (воскресенье = 0, понедельник = 1, ..., суббота = 6)

Для написания программы можно упростить формулу следующим образом:

можнопредставить в виде таблицы сдвигов месяцев . В итоге получится следующая таблица:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Месяц | Jan | Feb | Mar | Apr | May | Jun | July | Aug | Sep | Oct | Nov | Dec |
| Сдвиг | 0 | 3 | 2 | 5 | 0 | 3 | 5 | 1 | 4 | 6 | 2 | 4 |

После всех упрощений конечная формула примет вид:

# Текстовый алгоритм решения задачи

Таблица – Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  |  |
|  | YearShift = (0, 3, 2, 5, 0, 3,  5, 1, 4, 6, 2, 4) |
|  | Ввод Y |
|  | Month := January |
|  | Начало цикла А1. Проверка выполнения условия (Month ≤ December). Если условие истинно, идти к шагу 6, иначе – к шагу 18 |
|  | D := 1 |
|  | Проверка выполнения условия Month < March. Если условие истинно, идти к шагу 8, иначе – к шагу 10 |
|  | YTemp := YTemp – 1 |
|  | Идти к шагу 11 |
|  | YTemp := Y |
|  | Начало цикла А2 |
|  | WDay := ( D + YearShift[ord(Month)] + YTemp + YTemp div 4 - YTemp div 100 + YTemp div 400 ) mod 7 |
|  | D := D + 1 |
|  | Конец цикла А2. Проверка выполнения условия (WDay = 1). Если условие истинно, идти к шагу 15, иначе – к шагу 11 |
|  | Day[m] := D - 1 |
|  | Month := Succ(Month) |
|  | Конец цикла А1. Идти к шагу 5 |
|  | Month := January |
|  | Начало цикла А3. Проверка выполнения условия (Month ≤ December). Если условие истинно, идти к шагу 20, иначе – к шагу 22 |
|  | Вывод Day[ord(Month) + 1] |
|  | Month := Succ(Month) |
|  | Останов. |

# Структура данных

Таблица – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| WDay | Integer | День недели |
| D | Integer | День месяца |
| m | Integer | Месяц |
| Y | Integer | Год |
| YTemp | Integer | Временное значение года |
| Day | Array[1..12] of Integer | Массив дней месяцев, удовлетворяющих условиям задачи |
| Month | Пользовательский тип “Перечисление” | Месяц. Принимает значения = (January, February, March, April, May, June, July, August, September, October, November) |
| YearShift | Array[1..12] of Integer | Значения сдвига дней в месяце для любого года. Принимает значения = (0, 3, 2, 5, 0, 3, 5, 1, 4, 6, 2, 4) |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

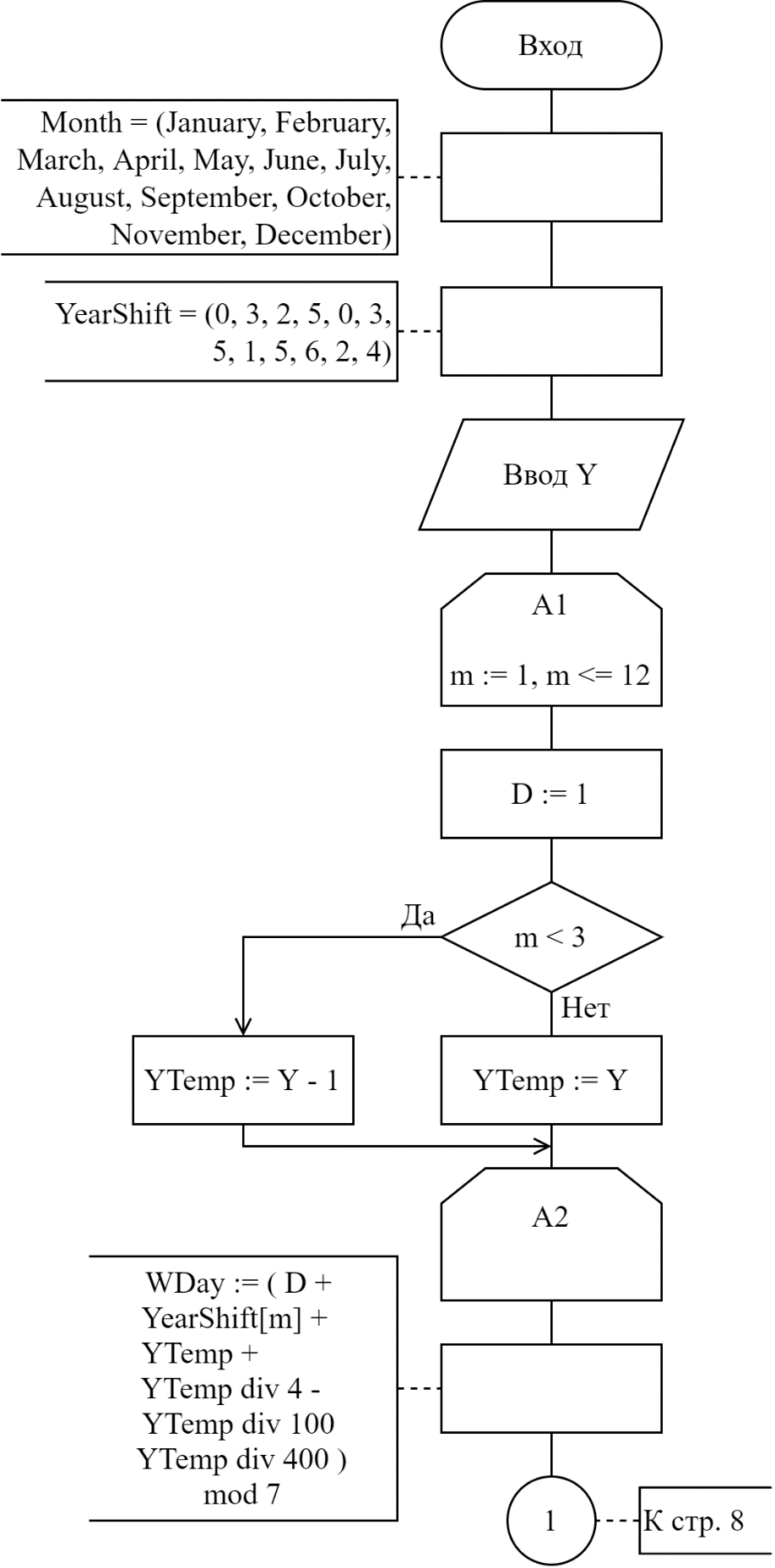


Рисунок - Схема алгоритма решения задачи по ГОСТ 19.701-90 (часть 1)

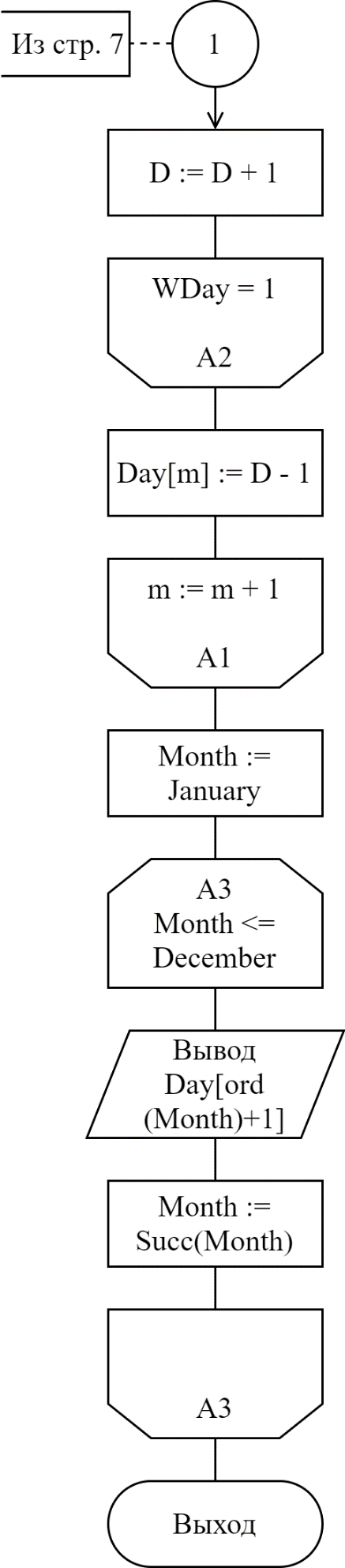


Рисунок - Схема алгоритма решения задачи по ГОСТ 19.701-90 (часть 2)

# Результаты расчетов

Вследствие выполнения программы на экран выводятся следующие результаты:



Рисунок 3 - Результаты расчётов

Приложение А

(обязательное)

program Lab6;

{

Find days of first monday in every month in given year

Input: Year

Output: Array of Days according to month

}

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

// TMonth - type of months in year

type

TMonth = (January, February, March, April,

May, June, July, August,

September, October, November, December);

// YearShift - shift in month according to disparate variation

// of Gauss's algorithm

const

YearShift: Array[1..12] of Integer =

(0, 3, 2, 5, 0, 3,

5, 1, 4, 6, 2, 4);

// WDay - day of week

// D - day

// m - month

// Y - year

// YTemp - temp value of year

// Day - array of days of fist Mondays

// Month - array of months' names

// InputStr - string to input value

// Error - value for errors in Val method

var

Error, m, D, WDay, Y, YTemp: Integer;

Day: Array[1..12] of Integer;

InputStr: String;

Month: TMonth;

begin

// Input Year greater than 0 using method Val

// Input string, take that string and convert

// it to integer

// If conversion fails pass index of failed char

// conversion to Error

// If Error is not equal to 0, repeat cycle

// If Inputted year is smaller than 0, repeat cycle

write('Enter year: ');

repeat

readln(InputStr);

val(InputStr, Y, Error);

if (Error <> 0) or (Y <= 0) then

writeln('Error! Enter a valid number.');

until (Error = 0) and (Y > 0);

// Iterate through month to find first Monday

// of month m

for m := 1 to 12 do

begin

// Set first day of month

D := 1;

// If month is January or December

// take previous year

if m < 3 then

YTemp := Y - 1

else

YTemp := Y;

// Find day of first monday in given month

// using disparate

// variation of Gauss's formula

// Iterate through days of month until you

// find first Monday

repeat

// Find day of week in given D of m

WDay := ( D + YearShift[m] +

YTemp +

YTemp div 4 -

YTemp div 100 +

YTemp div 400 ) mod 7;

// Go to the next day

D := D + 1;

until WDay = 1;

Day[m] := D - 1;

end;

// Output first mondays in months

// Iterate from January to December

for Month := January to December do

begin

// Output name of month

case Month of

January: write('January: ');

February: write('February: ');

March: write('March: ');

April: write('April: ');

May: write('May: ');

June: write('June: ');

July: write('July: ');

August: write('August: ');

September: write('September: ');

October: write('October: ');

November: write('November: ');

December: write('December: ');

end;

// Output day of first Monday

writeln(Day[ord(Month) + 1]);

end;

// Stop console from closing;

readln;

end.

Приложение Б

(обязательное)

Тестовые наборы

Тестовые наборы для проверки ввода данных

Тест 1

Тестовая ситуация: отрицательный год

Исходные данные:

-2020

Ожидаемый резуьтат:

Error! Enter a valid number.

Полученный результат:



Тест 2

Тестовая ситуация: нулевой год

Исходные данные:

0

Ожидаемый резуьтат:

Error! Enter a valid number.

Полученный результат:



Тест 3

Тестовая ситуация: вещественный год

Исходные данные:

2021.9

Ожидаемый резуьтат:

Error! Enter a valid number.

Полученный результат:



Тест 4

Тестовая ситуация: значение года не является числом

Исходные данные:

two thousand twenty one

Ожидаемый резуьтат:

Error! Enter a valid number

Полученный результат:



Тестовые наборы для проверки вычислений

Тест 1

Тестовая ситуация: заданный год – невисокосный

Исходные данные:

2021

Ожидаемый резуьтат:

4 1 1 5 3 7 5 2 6 4 1 6

Полученный результат:



Тест 2

Тестовая ситуация: заданный год – високосный

Исходные данные:

2020

Ожидаемый резуьтат:

6 3 2 6 4 1 6 3 7 5 2 7

Полученный результат:

