Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный технологический университет»

**Документация**

«Календарь»

Минск 2022

**Содержание**

**1. Программа календаря**

Я рассмотрел реализацию функции , которая выведет календарь на заданный пользователем год , путем вычесления дня недели 1 Января и применяя операторы Switch/Case.

**1.1. Алгоритм календаря**

Сначала в программе пользователь должен ввести целое число, представляющее собой день недели, соответствующий 1 января (0 — для понедельника, 1 — для вторника и так далее). Затем нужно ввести календарный год. После этого программа может напечатать заголовок календаря и использовать введенное значение года для создания флага високосного года leap\_year\_flag. При помощи операции деления по модулю (%) на 4 определяется остаток: ноль, если год — високосный, и ненулевое значение в противном случае (Код 1.2).

int jan\_1\_start\_day, num\_days\_per\_month, width = 0, month, date, leap\_year\_flag; // Укажите день недели 1 января

cout << "Please enter January 1's starting day;\n";

// 0 — указывает, что 1 января — воскресенье

cout << "\nA 0 indicates January 1 is on a Sunday,";

// 1 — указывает, что 1 января — понедельник

cout << "\nA 1 indicates January 1 is on a Monday,";

// 2 — указывает, что 1 января — вторник и т.д.

cout << "\nA 2 indicates January 1 is on a Tuesday, etc: ";

cin >> jan\_1\_start\_day;

// Укажите, на какой год вы хотите составить календарь:

cout << "\nEnter the year you want the calendar generated: ";

cin >> leap\_year\_flag;

// Календарь на год

cout << "\n\n The calendar for the year " << leap\_year\_flag; leap\_year\_flag = leap\_year\_flag % 4;

// признак високосного года (0)

Код 1.2

Далее 12 раз выполняется цикл, в котором печатается название текущего месяца и переменной num\_days\_per\_month присваивается соответствующее значение: число дней в данном месяце. Все это выполняется при помощи оператора switch, в котором проверяется текущее целое значение месяца month ( Код 1.3)

cout.width(20);

for (month = 1; month <= 12; month++) {

switch (month) {

case 1:

cout << "\n\n\n" << " January" << "\n"; // Январь

num\_days\_per\_month = 31;

break;

case 2:

cout << "\n\n\n" << " February" << "\n"; // Февраль

num\_days\_per\_month = leap\_year\_flag ? 28 : 29;

break;

case 3:

cout << "\n\n\n" << " March " << "\n"; // Март

num\_days\_per\_month = 31;

break;

case 4:

cout << "\n\n\n" << " April " << "\n"; // Апрель

num\_days\_per\_month = 30;

break;

case 5:

cout << "\n\n\n" << " May " << "\n"; // Май

num\_days\_per\_month = 31;

break;

case 6:

cout << "\n\n\n" << " June " << "\n"; // Июнь

num\_days\_per\_month = 30;

break;

case 7:

cout << "\n\n\n" << " July " << "\n"; // Июль

num\_days\_per\_month = 31;

break;

case 8:

cout << "\n\n\n" << " August " << "\n"; // Август

num\_days\_per\_month = 31;

break;

case 9:

cout << "\n\n\n" << "September" << "\n"; // Сентябрь

num\_days\_per\_month = 30;

break;

case 10:

cout << "\n\n\n" << " October " << "\n"; // Октябрь

num\_days\_per\_month = 31;

break;

case 11:

cout << "\n\n\n" << "November " << "\n"; // Ноябрь

num\_days\_per\_month = 30;

break;

case 12:

cout << "\n\n\n" << "December " << "\n"; // Декабрь

num\_days\_per\_month = 31;

break;

}

Код 1.3

Вслед за оператором switch, после печати названия месяца печатаются названия дней недели и пропускается некоторое количество пустых столбцов, в зависимости от того, на какой день недели пришелся первый день месяца (Код 1.4).

cout.width(0);

// Bc Пн Вт Ср Чт Пт Сб

cout << "\nSun Mon Tue Wed Thu Fri Sat\n";

cout << "--- --- --- --- --- --- --- \n";

for (date = 1; date <= 1 + jan\_1\_start\_day \* 5; date++)

cout << " ";

Код 1.4

И наконец, в последнем цикле for вычисляются и печатаются даты для каждого месяца. В последнем операторе программы вычисляется значение day\_code для следующего печатаемого месяца (Код 1.5).

for (date = 1; date <= num\_days\_per\_month; date++) {

cout.width(2);

cout << date;

if ((date + jan\_1\_start\_day) % 7 > 0)

cout << " ";

else

cout << "\n ";

}

jan\_1\_start\_day = (jan\_1\_start\_day + num\_days\_per\_month) % 7;

}

cin >> date;

Код 1.5

**2 Псевдокод программы**

Начало

Функция calendar()

Ввод и инициализация переменных jan\_1\_start\_day, num\_days\_per\_month, width = 0, month, date, leap\_year\_flag

Вывод

Please enter January 1's starting day ,

0 indicates January 1 is on a Sunday,

1 indicates January

1 is on a Monday,nA

1 indicates January

1 is on a Monday

Ввод

jan\_1\_start\_day;

Вывод

Enter the year you want the calendar generated

Ввод

leap\_year\_flag

Вывод

The calendar for the year, leap\_year\_flag

Ввести

leap\_year\_flag = leap\_year\_flag % 4;

cout.width(20);

Для месяца равного 1 и до 12 , увеличевая на 1

switch(month)

case 1

Вывод Январь

num\_days\_per\_month = 31;

case 2

Вывод Февраль

num\_days\_per\_month = leap\_year\_flag ? 28 : 29;

case 3

Вывод Март

num\_days\_per\_month = 31;

case 4

Вывод Апрель

num\_days\_per\_month = 30;

case 5

Вывод Май

num\_days\_per\_month = 31;

case 6

Вывод Июнь

num\_days\_per\_month = 30;

case 7

Вывод Июль

num\_days\_per\_month = 31;

case 8

Вывод Август

num\_days\_per\_month = 31;

case 9

Вывод Сентябрь

num\_days\_per\_month = 30;

case 10

Вывод Октябрь

num\_days\_per\_month = 31;

case 11

Вывод Ноябрь

num\_days\_per\_month = 30;

case 12

Вывод Декабрь

num\_days\_per\_month = 31;

Вывод

Sun Mon Tue Wed Thu Fri Sat

--- --- --- --- --- --- ---

Через каждые 5 дней

Вывод

" "

Пока date = 1 до num\_days\_per\_month увеличивая на 1

Ввод

cout.width(2);

Вывод

date;

Если (date + jan\_1\_start\_day) % 7 > 0)

Вывод

Пробел

Иначе

Следующая строка

Ввод

jan\_1\_start\_day = (jan\_1\_start\_day + num\_days\_per\_month) % 7;

Ввод

date;

Конец

**2.2 Блок-схема программы**



**2.3 Тестирование приложения**

Я провел тестирование приложения на корректность.

Тестирование показало , что календарь работает исправно при всех возможных способах вывода и ввода до 7 дня недели. При вводе более высокого числа произойдет смещение календаря на указзанное кол-во дней вправо.

В первом случае я ввожу , что январь это воскресенье и обозначаю , что это 2023 год, и если сверить с календарем то данный проноз будет верныт(Код 2.4).

Во втором случае январь уже будет понедельником и как видно на рисунке 2.5 то смещение будет верным. Что произойдет и в остальных случаях ввода до субботы(7).

В третьем случае я ввел значение дня недели 50 , что невозможно , в данном случае программа смещает календарь на 43 дня вправо (Код 2.6).



Код 2.4



Код 2.5



Код 2.6

**Приложение**

void calendar()

{

int jan\_1\_start\_day, num\_days\_per\_month, width = 0, month, date, leap\_year\_flag; // Укажите день недели 1 января

cout << "Please enter January 1's starting day;\n";

// 0 — указывает, что 1 января — воскресенье

cout << "\nA 0 indicates January 1 is on a Sunday,";

// 1 — указывает, что 1 января — понедельник

cout << "\nA 1 indicates January 1 is on a Monday,";

// 2 — указывает, что 1 января — вторник и т.д.

cout << "\nA 2 indicates January 1 is on a Tuesday, etc: ";

cin >> jan\_1\_start\_day;

// Укажите, на какой год вы хотите составить календарь:

cout << "\nEnter the year you want the calendar generated: ";

cin >> leap\_year\_flag;

// Календарь на год

cout << "\n\n The calendar for the year " << leap\_year\_flag; leap\_year\_flag = leap\_year\_flag % 4;

// признак високосного года (0)

cout.width(20);

for (month = 1; month <= 12; month++) {

switch (month) {

case 1:

cout << "\n\n\n" << " January" << "\n"; // Январь

num\_days\_per\_month = 31;

break;

case 2:

cout << "\n\n\n" << " February" << "\n"; // Февраль

num\_days\_per\_month = leap\_year\_flag ? 28 : 29;

break;

case 3:

cout << "\n\n\n" << " March " << "\n"; // Март

num\_days\_per\_month = 31;

break;

case 4:

cout << "\n\n\n" << " April " << "\n"; // Апрель

num\_days\_per\_month = 30;

break;

case 5:

cout << "\n\n\n" << " May " << "\n"; // Май

num\_days\_per\_month = 31;

break;

case 6:

cout << "\n\n\n" << " June " << "\n"; // Июнь

num\_days\_per\_month = 30;

break;

case 7:

cout << "\n\n\n" << " July " << "\n"; // Июль

num\_days\_per\_month = 31;

break;

case 8:

cout << "\n\n\n" << " August " << "\n"; // Август

num\_days\_per\_month = 31;

break;

case 9:

cout << "\n\n\n" << "September" << "\n"; // Сентябрь

num\_days\_per\_month = 30;

break;

case 10:

cout << "\n\n\n" << " October " << "\n"; // Октябрь

num\_days\_per\_month = 31;

break;

case 11:

cout << "\n\n\n" << "November " << "\n"; // Ноябрь

num\_days\_per\_month = 30;

break;

case 12:

cout << "\n\n\n" << "December " << "\n"; // Декабрь

num\_days\_per\_month = 31;

break;

}

cout.width(0);

// Bc Пн Вт Ср Чт Пт Сб

cout << "\nSun Mon Tue Wed Thu Fri Sat\n";

cout << "--- --- --- --- --- --- --- \n";

for (date = 1; date <= 1 + jan\_1\_start\_day \* 5; date++)

cout << " ";

for (date = 1; date <= num\_days\_per\_month; date++) {

cout.width(2);

cout << date;

if ((date + jan\_1\_start\_day) % 7 > 0)

cout << " ";

else

cout << "\n ";

}

jan\_1\_start\_day = (jan\_1\_start\_day + num\_days\_per\_month) % 7;

}

cin >> date;

}