DesignatingTheDayOfTheWeek

Software Documentation

Author: matiwa

Table of contents

Introduction

This software documentation includes: description of the application's operation, what is needed for use, algorithms used, interface description and source code description. This application is used to designate the day of the week.
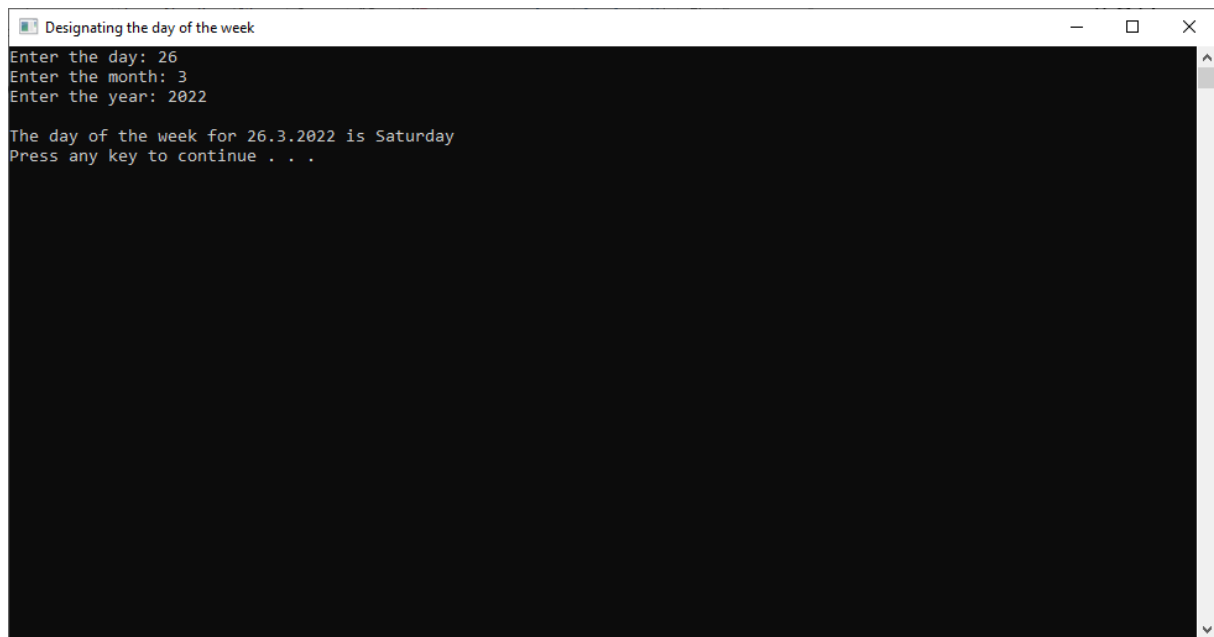
Describing of the application's operation

If the user wants to sort any non-empty set of numbers, run the *.exe file. After this operation, the console window will appear on the screen as below.
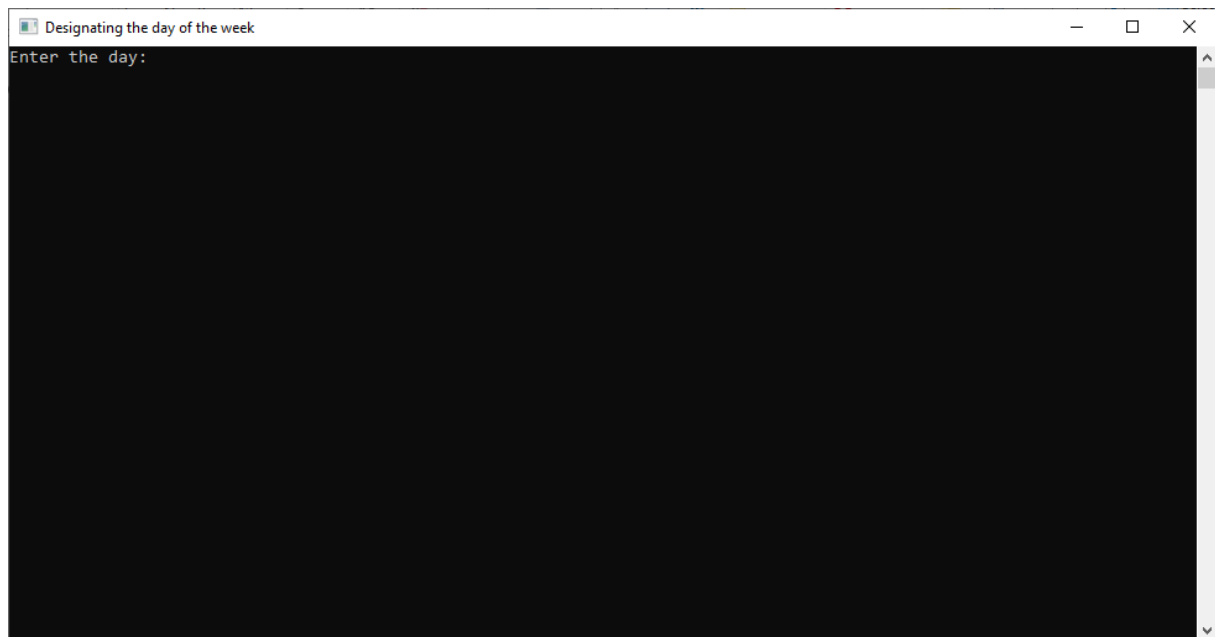
Drawing 1: The beginning of the application's operation [own study]

First, the user enters the number of the day of the week. Secondly, there is a month to be given (in the form of a number!). The user of the application then enters the year. After all, the result of the program is the name of the day of the week.

Drawing 2: The effect of the program [own study]

What is needed for use?

The application does not require installation. It only needs the Windows operating system.

Algorithms used

The algorithm used in the application is based on the designating the day of the week.

The algorithm presented here allows you to quickly calculate the day of the week for the given date day-month-year. All the division operations shown are integer operations - that is, they reject the remainder. So 5/2 = 2.

First, we calculate the day of the week for the first January of the given year. 0 means Monday, 1 Tuesday, ... 6th Sunday.

YY = (year-1) mod 100

C = (year-1) - YY

G = YY + YY / 4

day of the week for January 1 = (((((C / 100) mod 4) * 5) + G) mod 7)

Now we will define an auxiliary table specifying how many days of the year have passed for the given month. The table is defined for a non-leap year:

January	0
February	31
March	59

April	90
May	120
June	151
July	181
August	212
September	243
October	273
November	304
December	334

Then we will designate the day of the year:

- day of the year = the number of days which have passed for the given month taken from the table above + the day of the month
- Now it is still necessary to check whether one day should be added in connection with leap years. So the day of the year should be increased by 1 if the given month is after February and the given year is a leap year

We now have all the necessary data to set the day of the week. To the calculated day of the week for January 1, we add the designated day of the year minus 1 and divide the whole modulo 7. The result is the day of the week for the given date, 0 means Monday, 1 Tuesday, ... 6th Sunday.

day of the week = (day of the week for January 1 + day of the year - 1) mod 7

Example:

We will designate the day of the week for: 15-2-2008.

So let's first calculate the day of the week for January 1st.

YY = (year-1) mod 100 = (2008-1) mod 100 = 2007 mod 100 = 7.

C = (year-1) - YY = (2008-1) - 7 = 2007 - 7 = 2000.

G = YY + YY / 4 = 7 + 7/4 = 7 + 1 = 8.

day of the week for January 1 = (((((C / 100) mod 4) * 5) + G) mod 7) = (((((2000/100) mod 4) * 5) + 8) mod 7) = ((((20 mod 4) * 5) + 8) mod 7) = (((0 * 5) + 8) mod 7) = ((0 + 8) mod 7) = 8 mod 7 = 1.

So January 1, 2008 is Tuesday.

Now we calculate which day of 2008 is 15-2.

day of the year = 31 + 15 = 46.

The given year is a leap year, but the given month is not later than February, so do not add 1 to the day of the year.

So finally:

day of the week = (day of the week for January 1 + day of the year - 1) mod 7 = (1 + 46 - 1) mod 7 = 46 mod 7 = 4.

So 15-2-2008 is Friday. [1]

Interface description

The interface is a console pane. The operation of the program is based on user communication. He gives the needed values on the input. The course of the process and possible operating errors are described in the chapter „Describing of the application's operation".

Source code description

The project was made in the C++ programming language, in the Dev-C++ programming environment. All work was done on the Windows 10 operating system. The application's source code looks like this.

```cpp
#include<iostream>
#include<windows.h>
#include<conio.h>
using namespace std;

char *tydzien[] =
{"Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"};

int liczbaDni[] =
{0, 31, 59, 90, 120, 151, 181, 212, 243, 273, 304, 334};

int przestepny(int rok) {
    return ((rok % 4 == 0  &&  rok % 100 != 0) || rok % 400 == 0);
}

int dzienTygodnia(int dzien, int miesiac, int rok) {
    int dzienRoku;
    int yy, c, g;
    int wynik;

    dzienRoku = dzien + liczbaDni[miesiac-1];
    if ((miesiac > 2) && (przestepny(rok) == 1)) dzienRoku++;
```

```
        yy = (rok - 1) % 100;
        c = (rok - 1) - yy;
        g = yy + (yy / 4);
        wynik = (((((c / 100) % 4) * 5) + g) % 7);
        wynik += dzienRoku - 1;
        wynik %= 7;

        return wynik;
}
int main(){
        SetConsoleTitleA("Designating the day of the week");
        int d,m,y;

        cout<<"Enter the day: ";
        cin>>d;
        cout<<"Enter the month: ";
        cin>>m;
        cout<<"Enter the year: ";
        cin>>y;
        cout<<endl<<"The day of the week for "<<d<<"."<<m<<"."<<y<<" is
";
        cout<<tydzien[dzienTygodnia(d,m,y)]<<endl;
        system("pause");
        return 0;
}
```

Listing 1: Source code [own study]

List of drawings

List of listings

Bibliography

[1] http://www.algorytm.org/przetwarzanie-dat/wyznaczanie-dnia-tygodnia.html