Lab 8.2.1 Enums: representing values

Objectives

Familiarize the student with:

- representing values with enums;
- the modulo operator.

Scenario

Write a calendar class that uses enums to find out which month and day of the week it is. This class should have two methods: the first is a constructor, to which we can pass three parameters – day, month and year. The second is a print method, which prints the date and day of the week, and how many days have passed since the 1st of January 1970; the month and day of the week are represented by words e.g. if we pass (13, 3, 2016) to the constructor, the print method must print: 13 march 2016 - Sunday - 16872 days since 1st January 1970. Use enums in the print method to represent the days of the week and the months, and combine them with tables of strings (which hold the names of the days and months). If someone passes a date which is earlier than 1st January 1970, then the constructor must throw an exception. To count how many days have passed since the 1st of January 1970, we should sum up all the years multiplied by 365 (or 366 for leap years). To check if a year is a leap year, you should check if it is divisible by 4, not divisible by 100 or divisible by 400 (it's best to use the modulo operator: '%'). To obtain the day of the week, simply divide modulo 7. January 1, 1970, was a Thursday. If the result of dividing modulo 7 is 1, then the day of the week is a Friday. If it's 3, then the day of the week is a Sunday, and so on (6 is Wednesday and 0 is Thursday). Try to write your code in the shortest possible way. Write test code to get input from the user and print one piece of date information in one program run.

Example input

```
13
3
2016
```

Example output

```
13 march 2016 - Sunday - 16872 days since 1st January 1970
```