Lab 3.4.2 One step further: finding the lengths of months

Objectives

Familiarize the student with:

- building a set of cooperating functions,
- singnalling erroneous arguments using a specific return value.

Scenario

Let's continue our coder's reflections on time. Now, when you have a reliable function diagnosing the nature of any year, you can use it to implement another important function returning the length of any month (measured in days, of course).

Write a **function** equipped with the following features:

- its name is "monthLength"
- it accepts two arguments of type int: year number (first) and month number (second)
- it returns an **int** value which represents a length of specified month in a specified year (obviously, year is important only when month == 2) or 0 if any of the input arguments isn't valid
- it should be *mute*

We've prepared a skeleton of the program - fill the function body with an appropriate content!

We've also attached the output that is expected from your program.

Hint: there are at least two ways of implementing the function: you can use **switch** or (something which seems a bit smarter) declare a vector storing months' lengths – choose the more convenient style.

```
#include <iostream>

using namespace std;

bool isLeap(int year) {

 // The code you've inserted already
}

int monthLength(int year, int month) {

 // Insert a new code here
}

int main(void) {

 for(int yr = 2000; yr < 2002; yr++) {
  for(int mo = 1; mo <= 12; mo++)
   cout << monthLength(yr,mo) << " ";
  cout << endl;
 }
 return 0;
}
```

Example output

```
31 29 31 30 31 30 31 31 30 31 30 31
31 28 31 30 31 30 31 31 30 31 30 31
```