

Homework 1

Assigned Tuesday, February 1. The homework must be uploaded to uzak.etu.edu.tr by **Saturday, February 12** anytime upto midnight (23.59). Before submitting, place all your files in a directory called “YourFullName-HW1”, **tar and gzip** your directory, and upload a single file.

In this assignment, you will write a program that displays a Gregorian calendar for any year/month. Essentially, you will be implementing the Linux ‘cal’ command.

Actually the Gregorian calendar was not adopted in Europe until 1582, and not until 1752 in Britain (If you check with the Linux ‘cal’ command, you will see that the 11 days between September 3,1752 and September 13, 1752 are missing.) The Gregorian calendar did not become the world standard until the 1900 on. For the purpose of this program though, we are still going to display a Gregorian calendar for every year 1-9999.

Here’s an algorithm for calculating the calendar:

1. Find the day of the week a year starts on. In general, for every year that is not a leap year, next year starts one day later in the week. For every year that is a leap year, next year starts two days later in the week. 2022 starts on a Saturday. Count backwards (subtract) for any year before 2022 following the same rule.

Rules for deciding a leap year:

- Every year divisible by 4 is a leap year.
- However, every year divisible by 100 is not a leap year.
- However, every year divisible by 400 is a leap year after all.

2. Determine the number of days in a month.
3. Now that you have the day of the week the year starts on, and the number of days in any month, you can find out the day of the week each month starts on. Indent (print white spaces) accordingly, and you have the correct calendar.

Your calendar program should provide the options as described below.

- Running the program with the flag **-m month**, will print the calendar for the given month of the current year. The month argument may be given as a number between 1 and 12, or as a word like January or january, or as an abbreviation like Jan or jan.

```
$hw1 -m Jan
      January 2022
Su Mo Tu We Th Fr Sa
                1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
```

- Running the program with the flag **-y year**, will prints the entire calendar of the given year. Run `'cal 2022'` to see the expected format.
- **-y** and **-m** flags may be used together. Also these flags can come in any order, i.e. **-m 2 -y 2022** is equivalent to **-y 2022 -m 2**. For example the run below will print the calendar of February 2022.

```
$hw1 -m 2 -y 2022
```

- Running the program with the flag **-h**, will print out a Unix-like manual page informing the user how to use your program and its arguments. Also print this screen whenever improper input has been detected. Also, if there is a **-h** flag, it cancels out all others, i.e. help screen will be printed and nothing else.
- Any or all flags may be omitted. If no flags are given, default is **-h**.

Other requirements:

- You must format your output so that the days of the week line up. Run `'cal 2022'` to see the expected format. For more information on printf formatting, refer to the lecture slides or the text book.
- Pay special attention to designing your functions and your program organization. Your program will be graded on its organization and modularity as well as functionality. Some suggested functions (but you probably need more than these):
 - A function that takes a year and decides if it is a leap year or not
 - A function that takes a year and finds the day of the week on which the given year starts.
 - A function that determine the number of days in a given month.
 - A function that prints the calendar for a given month.
 - A function that prints the calendar for a given year
- Provide a prototype for every function in your program in a header file, which you then include in your source file using the `#include` directive. The file should be named the same as your program but with extension `.h`. Submit it together with your source code.