

## CSI 333 – Programming at the Hardware-Software Interface – Fall 2011

### Programming Assignment I

**Date given:** Sep. 8, 2011

**Due date:** Sep. 16, 2011

**Weightage:** 5%

This is an *individual* assignment. The regular deadline for this assignment is **11 PM, Friday, September 16, 2011**. With lateness penalty, the program will be accepted until **11 PM, Sunday, September 18, 2011**. The assignment will *not* be accepted after that deadline.

**Important notes:** Your C source program for this assignment must be in a file called `p1.c`. Instructions regarding how your source file can be submitted electronically will be provided in the `README` file for this assignment. You must follow the programming and documentation guidelines indicated in the handout on “Course Policies”.

---

The total grade for the assignment is 100 points, with 90 points for correctness and 10 points for documentation.

You are required to write a C program that accepts two dates in the year 2011 and prints out the number of days which are *strictly between* the two specified dates. Each date will be specified by two integers, with the first integer representing the month and the second representing the day.

**Example:** The two integers 2 and 10 represent the date February 10, 2011. Likewise, the two integers 3 and 7 represent March 7, 2011. The number of days that lie strictly between these two dates is 24. (Of these, 18 are in February and the remaining 6 are in March.) As another example, if the first date is specified by the integers 10 and 17 (which represent the date October 17, 2011) and the second date is specified by the integers 10 and 18 (which represent the date October 18, 2011), the answer to be printed is 0.

The outline for your program is as follows.

1. Prompt the user for the first date. Read the date and check if it is valid. If it is not valid, print a suitable error message and stop. (Conditions for the validity of a date are given later in this handout.)
2. Prompt the user for the second date. Read the date and check if it is valid. If it is not valid, print a suitable error message and stop.
3. If the second date *precedes* the first date, print a suitable error message and stop.
4. Compute the number of days which are strictly between the first and the second dates.
5. Print the answer computed in Step 4 and stop.

**Conditions for a date to be valid:** Suppose a date is specified by the two integers  $m$  (the month) and  $d$  (the day). For this date to be valid, *both* of the following conditions must be satisfied.

- (a) The value of  $m$  should be at least 1 and at most 12.
- (b) The value of  $d$  should be at least 1 and at most the maximum number of days possible in the month specified by  $m$ .

**Additional notes about the program:**

- (a) For this assignment, your program may consist of just one function, namely `main`.
- (b) Your program should read all its input from `stdin`; that is, you must use the `scanf` function for reading the input.
- (c) Your program must write all its output to `stdout`; that is, you must use the `printf` function for producing prompts, error messages and the final result.
- (d) You may assume that for each prompt, the user will type two integers separated by one or more spaces.
- (e) Other than the validity checks mentioned in Steps 2 through 4 in the program outline above, no other error checks are needed.

---

**Information about README file:** The README file for this assignment will be available by 10 PM on Saturday, September 10, 2011. The name of the file is `prog1.README` and it will be in the directory `~csi333/public/prog1` on `itsunix.albany.edu`. This file will contain information about how you can electronically submit your source file `p1.c`. It may also include information regarding some sample inputs and outputs that you can use to test your program.