

**CS 1344 Fall 96**  
**Assignment 1: Calendar Week**  
**Due Date: Thur Sept 26**

---

Problem

Output a weekly calendar for a given date. The calendar is to be displayed on both the screen and redirected to a text file (*weekly.txt*).

Discussion

The program is only to work correctly for weeks in years >1752. Note - it was in 1752 that the change from the Julian to the Gregorian calendar occurred. Thus the program will be designed only for the Gregorian calendar. The program must handle leap years correctly. A leap year occurs every fourth year (1984, 1988 ...), provided the year is not a century, then it is every 400 years. For example, 1800 and 1900 were not leap years, but 2400 will be a leap century. In order to determine what day of the week any given date falls upon, the following formula may be used:

```
calcdte  ->   day + 2 * month + 3 * (month + 1) DIV 5 + year  
            + year DIV 4 - year DIV 100 + year DIV 400 + 1
```

```
weekday  ->   calcdte MOD 7
```

```
CASE      weekday  
          0 -> Sunday  
          1 -> Monday  
          :  
          :  
          6 -> Saturday
```

The above formula requires that January and February be treated as the 1<sup>st</sup> & 14<sup>th</sup> months of the preceding year.

Execution

The program should initially present the user with a startup screen displaying the name of the program, a brief 3-4 line explanation of the program and the programmer's name, and email address. Minor creativity in the presentation of the startup screen is allowed, (but not required). After waiting until the user hits the return key the startup screen is to be cleared. A brief help screen should appear explaining the program to the user, which is cleared upon the pressing of the return key.

Following the brief help screen the user is prompted for the desired date of the week calendar. The user's input must be checked for *correctness*. From the appendix on *Program Testing and Correctness*, number 4 is the minimum acceptable level for this program, but level 6 is the expected level. When an error occurs it must be explained to the user, who is then prompted again for the desired month, day and year. A vertical week calendar is then output that contains the date the user has input, (see following output section). After the week calendar is displayed, the user must be given the option to quit or requesting another calendar week for display. The program should issue a brief termination message before returning to the operating system.

Input

Output the week calendar for the following dates: 9/26/1996, 2/29/1996, 2/28/1995, 12/31/1996, 2/1/2000, 12/13/1957 and the date of your birth. Use the following data to test for illegal input: 1/1/1752 , 0/0/1996 , 13/13/1996 , -12/12/1996,

2/29/1995, and 12/12/-1996. Turn in the output only for the above data, (i.e. only turn in hard/soft-copies of two files: source code listing and the redirected program output file, "weekly").

### Assumptions

It may be assumed that the user will not enter character data when numeric input is requested, but this is the only assumption about the user input that may be made. Invalid user input must be checked and appropriate error messages issued. Operating System or compiler error messages are unacceptable.

### Output

The week calendar output on the screen and in the text file should closely resemble the following format.

**Week for: 9/26/1996**

S 22	
M 23	
T 24	
W 25	
H 26	
F 27	
S 28	

Note: for week calendars that occur across months, (or years), only the number of the preceding or following (legal) days must be output, (i.e. the changing month or year number need not be displayed).

### Grading

Turn in hard copies of the source code, input/output files and a top-down hierarchical design tree of the program. In addition, submit a diskette with files containing: ASCII source code, executable image, I/O files, and an ASCII *readme* file with execution instructions. The source code file must be named week.cc and the executable image week.exe. The disk should contain no subdirectories. All materials to be graded must be placed in a sealed folder, neatly labeled with your name, SSN, course number, index number & date.

In addition, your GTA may require you to demonstrate your program. To receive partial credit for programs that are non-working, or are not fully functional, a brief one or two paragraph description of the problem(s) must be included in the assignment folder. The location, routine minimum, must also be specified along with possible corrections that need to be made.