CP2 Exercise 1 **23/10/2006**

```
/* session 1 exercise 1 framework - Eike Anderson, 2005 */
/* compile as  > gcc -ansi -pedantic ex0101.c -o ex0101 */
#include<stdio.h>

int  isLeapYear(int year);
/* prototype for leap year function */

void printWeekday(int day,int month,int year);
/* prototype for weekday printing function */

int main(void)
{
  int day,month,year;

  printf("please enter a date in the format DD/MM/YYYY - ");
  scanf("%d/%d/%d",&day,&month,&year);
  printf("\nThe %02d/%02d/%04d is a ",day,month,year);
  printWeekday(day,month,year);
  printf("\n");

  return 0;
}

int isLeapYear(int year)
{
  if(year%4==0)
  {
    if(year%100==0)
    {
      if(year%400==0) return 1;
      else return 0;
    }
    else return 1;
  }
  return 0;
}

void printWeekday(int day,int month,int year)
{
  /* insert function body here */
}
```

1. The Gregorian calendar was introduced in the British Empire in the year 1752. To compensate for errors, 11 days were dropped from September 1752 (*September 2 was followed by September 14*). The first of January 1753 was a Monday. Expand the above program framework by completing the function

```
void printWeekday(int day, int month, int year);
```

The function should print out the name of the weekday of the date (*after 01/01/1753*) specified by the parameters "day", "month" and "year". You may need to write additional functions to complete this exercise.

2. Expand the previous exercise by dvising an additional function:

```
int dateExists(int day, int month, int year);
```

This function should return ***true*** (*1*) if the given date is valid or ***false*** (*0*) if the date is invalid. A valid date would be "05/11/1976" - an invalid date would be "30/02/1991". Modify the program so it rejects invalid dates and asks for valid ones instead before trying to calculate the weekday.

3. Based on the program created in the previous exercise design and implement a program that prints out a calendar listing for a given month:

Run-time example:

```
10 2006
Mon       Tue       Wed       Thu       Fri       Sat       Sun
                                                            1
2         3         4         5         6         7         8
9         10        11        12        13        14        15
18        19        20        21        22        23        24
23        24        25        26        27        28        29
30        31
```

Modify the function for printing the weekday from the previous exercise so that it returns a value between (*and including*) 0 (*Sunday*) and 6 (*Saturday*) instead of printing the weekday:

```
int weekday(int day, int month, int year);
```

Once you have done this write a function for printing the listing for the month

```
void printMonth(int month, int year)
{
  printf("%02d %04d\n",month,year);
  printf("Mon\tTue\tWed\tThu\tFri\tSat\tSun\n");

  /* complete function body here */
}
```

This function should print out a calendar listing for the month (*after the year 1752*) specified by the two parameters "month" and "year".