/* Coding Problems (25 points) */

/*

Please write code for the two problems below. Write the code for both problems in functions below the main function and call both from the main function. The functions should accept data from main, calculate and return the result to main (Problem 2 should print the results for the substance to screen. See below.). Getting data from the user and printing the results should be done in main. When completed, the main function should get input data from the user, call the functions, receive the results of the function calls, and print their results to screen.

1. Write a function that determines the day number (1 to 366) in a year for a date that is provided as input data. As an example, January 1, 1994, is day 1. December 31, 1993, is day 365. December 31, 1996, is day 366, since 1996 is a leap year. A year is a leap year if it is divisible by four, except that any year divisible by 100 is a leap year only if it is divisible by 400. Your function should accept the month, day, and year as integers from main as arguments. Include an additional function leap that returns 1 if called with a leap year, 0 otherwise.
Sample screen output:

```
CALCULATE DAY OF YEAR
Enter a date: 12 31 2004
12/31/2004 is the 366th day of 2004
```

Note that a leap year is a year that is divisible by 4 but not by 100 or divisible by 400. 1900 was not a leap year because it's divisible by 4 but is also divisible by 100. 2000 was a leap year because it's divisible by 400. 1904 was a leap year because it's divisible by 4 but not divisible by 100.

2. The table below shows the normal boiling points of several substances. Write a function that accepts an argument the observed boiling point of a substance in degrees Celsius and identifies the substance if the observed boiling point is within 5% of the expected boiling point. If the data input is more than 5% higher or lower than any of the boiling points in the table, the function should output the message Substance unknown. Note that the function should print the substance and not main. This will simplify the function call in main. You need to test each substance in an else if chain for the temperature passed from main.

Substance	Normal boiling point (Celsius)
Water	100
Mercury	357
Copper	1187
Silver	2193
Gold	2660

Your function should define and call a function within_x_percent that takes as parameters a reference value ref , a data value data , and a percentage

```
value
    x and returns 1 meaning true if data is within x % of ref -that is, (ref - x%
    * ref) <= data <= (ref + x % * ref) . Otherwise within_x_percent would
    return zero, meaning false. For example, the call within_x_percent(357,
    323, 10) would return true, since 10% of 357 is 35.7, and 323 falls between
    321.3 and 392.7. Sample screen output:

        FIND SUBSTANCE FOR BOILING POINT
        Enter a temperature: 360
        The substance is Mercury

*/

// Preprocessor directives
#include <stdio.h>
#define BOILING_POINT_WATER 100
#define BOILING_POINT_MERCURY 357
#define BOILING_POINT_COPPER 1187
#define BOILING_POINT_SILVER 2193
#define BOILING_POINT_GOLD 2660
#define DEVIATION_PERCENT 5

// Function prototypes
int IsLeapYear(int YearToCheck);
int DaysInYear(int year, int month, int day, int IsLeapYear);
int within_x_percent(int referenceTemperature, float actualTemperature, int
deviationPercent);
void FindAndPrintSubstance(float UserEnteredTemperature, int VariationPerent);

// Main function
int main()
{

printf("\n CALCULATE DAY OF YEAR.");
int userYear, userMonth, userDay = 0;

//Getting the Year from the user
printf("\n Please enter the year (YYYY): ");
scanf("%d", &userYear);

//Getting the Month from the user
printf("\n Please enter the month (MM): ");
scanf("%d", &userMonth);

//Getting the Day from the User
printf("\n Please enter the day (DD): ");
scanf("%d", &userDay);

int LeapYear = IsLeapYear(userYear);
if(LeapYear == 1)
{
    printf("This is a leap year.");
}
else
{
    printf("This is not a leap year.");
}

//calculate the number of days till date
```

```
int DaysTillDate = 0;

DaysTillDate = DaysInYear(userYear, userMonth, userDay, LeapYear);

printf("\n %02d/%02d/%04d is the %dth day of %04d.", userMonth, userDay, userYear,
DaysTillDate, userYear);


printf("\n FIND SUBSTANCE FOR BOILING POINT:");
float UserEnteredTemperature = 0.00;
printf("\n Please enter a temperature in degrees Celcius:");
scanf("%f", &UserEnteredTemperature);
FindAndPrintSubstance(UserEnteredTemperature, DEVIATION_PERCENT);

return 0;
}


// Function for Coding Problem 1
int DaysInYear(int year, int month, int day, int IsLeapYear)
{
        //calculate the number of days till date.
    int NumberOfDaysTillDate = 0;


    //first add the days
    NumberOfDaysTillDate = NumberOfDaysTillDate + day;


    for(int loopMonth = 1; loopMonth < month; loopMonth++)
    {
        //Figuring out odd months
        if(loopMonth % 2 != 0)
        {
                //Odd months have 31 days
                NumberOfDaysTillDate = NumberOfDaysTillDate + 31;
        }
        else
        {
            //Figuring out even months
            if(loopMonth == 2)
            {
                //Feburary has 28 days
                NumberOfDaysTillDate = NumberOfDaysTillDate + 28;
                if (IsLeapYear == 1)
                {
                    //add one day for leap year
                    NumberOfDaysTillDate = NumberOfDaysTillDate + 1;
                }
            }
            else
            {
                //Even month but not Feburary (30)
                NumberOfDaysTillDate = NumberOfDaysTillDate + 30;
            }

        }
    }
return NumberOfDaysTillDate;
```

```
}

// leap function
 int IsLeapYear(int YearToCheck)
{
    int IsGivenYearALeapYear = 0;
    if((YearToCheck%400 == 0) || ((YearToCheck%4 == 0) && (YearToCheck%100 != 0)))
    {
        IsGivenYearALeapYear = 1;
    }

    return IsGivenYearALeapYear;
}

// Function for Coding Problem 2


// within_x_percent function
int within_x_percent(int referenceTemperature, float actualTemperature, int
deviationPercent)
{
    int IsTemperatureWithinDeviationRange = 0;
    float DeviationValue = (referenceTemperature * deviationPercent)/100.00;
    if(((referenceTemperature - DeviationValue) <= actualTemperature) &&
(actualTemperature <= (referenceTemperature + DeviationValue)))
        {
            IsTemperatureWithinDeviationRange = 1;
        }
    return IsTemperatureWithinDeviationRange;
}

//Print the Substance
void FindAndPrintSubstance(float UserEnteredTemperature, int VariationPerent)
{
    if(within_x_percent(BOILING_POINT_WATER, UserEnteredTemperature,
VariationPerent) == 1)
    {
        printf("This substance is water.\n");
    }
    else if(within_x_percent(BOILING_POINT_MERCURY, UserEnteredTemperature,
VariationPerent) == 1)
    {
        printf("This substance is mercury.\n");
    }
    else if(within_x_percent(BOILING_POINT_COPPER, UserEnteredTemperature,
VariationPerent) == 1)
    {
        printf("This substance is copper.\n");
    }
    else if(within_x_percent(BOILING_POINT_SILVER, UserEnteredTemperature,
VariationPerent) == 1)
    {
        printf("This substance is silver. \n");
    }
    else if(within_x_percent(BOILING_POINT_GOLD, UserEnteredTemperature,
VariationPerent) == 1)
    {
        printf("This substance is gold. \n");
    }
```

```
    else
    {
        printf("This substance is unknown. \n");
    }
}
```