Before you begin your work, please create a new file folder on your computer. The name of the folder should be YourLastName_YourFirstName For example, if your name is John Smith your folder should be named Smith_John. Please store all your C files in that folder. Name your C files simply 1.c, 2.c, 3.c, 4.c, and 5.c.

At the end of the exam you must copy that folder onto a USB stick. Also, you must e-mail your files to the instructor.

Please attach all of your *.c files to one e-mail.
Use this subject line: "CprE185: Midterm 1, Section X"
Where X is your lab section (ask your TA if you don't know it).

Please DO NOT leave the room until you have done these 2 things:

- **copied your folder onto the USB memory stick**
- **and e-mailed your *.c files!!!**

1. HelloWorld in HTML (10 points)

The language that is used to create web pages is called HTML. The acronym stands for HyperText Markup Language. The html code is stored in *.html files, just like C is stored in *.c files. Each HTML file is a text file that gives instructions to a web browser on how to layout the web page that is described in the HTML file. The following HTML code is the equivalent of the Hello World program in C.

```
<!DOCTYPE html>
<html>
	<head>
		<title>Hello HTML</title>
	</head>
	<body>
		<p>"Hello World!"</p>
	</body>
</html>
```

Your task is to write a complete C program that prints the text of the above HTML program on the screen. Use tab characters to indent the lines.

Note: In case you are curious, if you store the HTML code given above in a file called HelloWorld.html and then try to open it with your web browser you will get the expected result: a web page that says Hello World!

2. Escape Velocity (15 points)

In physics, the escape velocity is the speed at which a body (e.g., a rocket) must travel in order to escape the gravitational pull of a planet. The formula for the escape velocity is given below

$$v_e = \sqrt{\frac{2GM}{r}}$$

where G is the gravitational constant **($G=6.67\times10^{-11}\ m^3\ kg^{-1}\ s^{-2}$)**, M is the mass of the planet in kilograms, and r is the radius of the planet in meters.

Write a complete C program that prompts the user to enter the mass and the radius of the planet and prints the escape velocity on the screen. The program must define a function called calcEscapeVelocity with the appropriate number and type of arguments to do this calculation. The result must then be printed in the main function.

Hint: Vary large numbers can be entered from the keyboard using the "e notation". For example **$2x10^{30}$** can be entered as 2e30 at the prompt. This notation can also be used to initialize variables. For example, double a=2e3; is equivalent to double a=2000;

```
========== Sample run #1 (Earth) ===========
Enter the planet's mass (in kg): 5.9736e24
Enter the planet's radius (in km): 6400
The escape velocity is 11.2 km/s
============================================

========== Sample run #2 (Sun) =============
Enter the planet's mass (in kg): 1.989e30
Enter the planet's radius (in km): 695500
The escape velocity is 617.7 km/s
============================================
```

3. The @@ operator (15 points)

Write a C program that performs the '@@' operation, which is defined as follows:

```
  a @@ b = (a*a + 2*b) / (3*a*b + 2*(a-b))
```

Have the user enter two doubles called X and Y and perform the '@@' operation on the following combinations of X, Y, and 1:

```
  X @@ 1
  X @@ X
  X @@ Y
  Y @@ 1
  Y @@ Y
  Y @@ X
```

Remarks: Please replicate the exact text of the prompts and the output. Make sure the results of the '@@' operation are printed with two digits of precision (i.e., "%.2f" in printf). Even though the text must be the same, the numbers will be different depending on the user input.

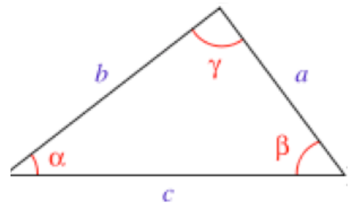
Important Note: The '@@' operator is fictitious and is not a native C operator. Also, it is not a valid character in the name of a variable or function.

```
============= START OF SAMPLE RUN =======================
Input X:
2.5
Input Y:
2.0
The results of '@@' operation:

X @@ 1 = 0.79
X @@ X = 0.60
X @@ Y = 0.64

Y @@ 1 = 0.75
Y @@ Y = 0.67
Y @@ X = 0.64
=============  END OF SAMPLE RUN  =======================
```

4. Law of Cosines (15 points)

The law of cosines is a very important formula in geometry. If you know the length of two sides of a triangle and the angle between those sides, then you can use the law of cosines to find the length of the third side of the triangle. The law is stated as follows:

$$c^2 = a^2 + b^2 - 2ab\cos\gamma$$



Where c is the length of the third side, and gamma is the angle between sides a and b.

Your assignment is to write a program that does the following:

- Asks the user to enter the length of two sides a and b
- Asks the user to enter the angle between a and b(in degrees)
- Calls a function to calculate and return the length of the third side, but this function must first call another function to convert the angle from degrees to radians.
- Finally, the main function must print the length of c.

```
============= START OF SAMPLE RUN =======================
Enter length of side a: 4.5
Enter length of side b: 9.7
Enter angle between a and b, in degrees: 52.5

The length of side c is 7.82.
=============  END OF SAMPLE RUN  =======================
```

5. Piano Keys (15 points)

The modern piano usually contains 88 keys that produce sound at frequencies between 27 and 4200 Hz. For example, key number 49 plays the note A4, which corresponds to sound at 440 Hz. The frequency f of a key can be computed from its number n using the following exponential function:

$$f(n) = 440 \times 2^{((n-49)/12)}$$

Your goal is to solve the inverse problem: given a frequency (f) compute the number for the key that produces sound with the frequency closest to f. In addition to the number, your program also has to print the number of the octave in which the key falls (i.e., octave 0, or octave 1, or octave 2, etc.)

The number of the octave o can be computed from the number of the key n using the following C expression:

```
o = (n + 8) /12,
```

where both o and n are assumed to be integers.

Hint: To obtain n from f you can use the following equation:

$$n = 12 * \log_2 (f/440) + 49$$

Because the key number must be an integer, you will need to round the value of the right-hand side of the above equation to the nearest integer. Thus, 49.3 rounds to 49 and 49.6 rounds to 50. Selecting the mode for rounding the exact halves is up to you. In other words, 49.5 may round to 50 or to 49.

Remember that assigning a floating point value to an integer in C simply truncates its fractional part, which is not what is needed here.

Hint 2: The standard mathematical function log() computes the natural logarithm, i.e., logarithm base e. You will need to use the change of base formula, i.e., divide the value returned by log() by the value of log(2.0).

```
============= START OF SAMPLE RUN =======================
Enter the key frequency> 440
The piano key that produces frequency closes to 440 is key number 49.
This key is in octave 4.
=============  END OF SAMPLE RUN  =======================

============= START OF SAMPLE RUN =======================
Enter the key frequency> 553
The piano key that produces frequency closes to 553 is key number 53.
This key is in octave 5.
=============  END OF SAMPLE RUN  =======================

============= START OF SAMPLE RUN =======================
Enter the key frequency> 35.900
The piano key that produces frequency closes to 35.9 is key number 6.
The key is in octave 1.
=============  END OF SAMPLE RUN  =======================
```