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. Averages (10 points)

Write a complete C program that takes in three integers from the user and computes the average of the first two, the second two, and the first and third number. It then calculates the average of all three numbers. The program must take the user input as integers (not doubles) and must replicate the following sample run exactly. Note where extra spaces are included. Also note that the 'and' and 'is' in the lines for the first three averages line up.

You can assume that the user will enter only numbers in the range 0-99.

```
============ START OF SAMPLE RUN ======================
Enter the first  number > 5
Enter the second number > 12
Enter the third  number > 21

The average of  5 and 12 is  8.500.
The average of 12 and 21 is 16.500.
The average of  5 and 21 is 13.000.
The average of  5, 12, and 21 is 12.667.
=============  END OF SAMPLE RUN  ======================
```

2. Apples (15 points)

Bill owns a small apple orchard. Every Saturday, he sells baskets of apples at the local market exclusively to his neighbors Sally and Jeff.

The apples are sold in the following manner:

1. Bill reserves 2 baskets for Jeff.
2. Bill sells 3/5 of his baskets to Sally (rounded down to the nearest basket).
3. Bill sells the remaining baskets plus the reserved baskets to Jeff.

Assume Bill always has at least 2 baskets to sell.

Write a program for Bill that calculates how many baskets Sally and Jeff will buy and how much they will owe him. Prompt the user to enter the number of baskets available and the price per basket. Make sure the cost is printed with two digit precision (i.e., "%.2f" in printf).

```
============ START OF SAMPLE RUN ======================
Number of baskets: 12
Price per basket: 5.35

Sally will buy 6 baskets for $32.10
Jeff will buy 6 baskets for $32.10
============  END OF SAMPLE RUN  ======================
```

3. Great Circle Distance (15 points)

Because the Earth is a sphere, the shortest distance between two points on the Earth's surface is not a straight line as you would draw on a map, but rather an arc. This arc is called the great circle route. This is why flights from Los Angeles to London will fly over Greenland and not the Eastern US. The great circle distance between two locations is given by this formula:

```
d = R*acos(sin(lat1)*sin(lat2) + cos(lat1)*cos(lat2)*cos(lon2 - lon1))
```

Where:

* d is the great circle distance between the 2 locations
* lat1 and lat2 are the latitudes of locations 1 and 2, respectively
* lon1 and lon2 are the longitudes of locations 1 and 2, respectively
* R is the radius of the Earth (use 6371 km)
* sin, cos, and acos are the sine, cosine, and arccosine functions

You must write a complete C program that reads in the latitudes and longitudes of the two locations (in degrees) and outputs the great circle distance (in km) on the screen. All of the relevant mathematical functions can be found in math.h. Latitude is positive for North and negative for South, and Longitude is positive for East and negative for West.

Print the result to 2 decimal places, i.e. `%.2f` in printf.

```
============= START OF SAMPLE RUN ======================
Enter the coordinates of location 1 (lat lon): -23.5 -46.62
Enter the coordinates of location 2 (lat lon): 41.9 12.5

The great circle distance between the locations is 9471.56 km.
=============  END OF SAMPLE RUN  ======================
```

4. Passing Efficiency Calculator (15 points)

Write a C program that calculates passer rating for NCAA quarterbacks.

The Inputs:

Pass Attempts - ATT
Pass Yards - YDS
Completions - COMP
Passing Touchdowns - TDS
Interceptions - INTS

The output is the passer rating, also called passing efficiency.

All inputs should be whole numbers.
The output should be truncated at 2 places after the decimal point.

The following is the relevant equation:

Passer Rating = ((8.4*YDS)+(330*TDS)+(100*COMP)-(200*INT))/ATT

The formula calculation must be performed inside a function that returns the passer rating to the main function, which prints it.

The sample run below uses stats from ISU quarterback Steele Jantz' 2011 season.

```
============= START OF SAMPLE RUN ======================
Enter Pass Attempts: 259
Enter Passing Yards: 1519
Enter Touchdowns: 10
Enter Completions: 138
Enter Interceptions: 11

This quarterback has a passer rating of 106.79
=============  END OF SAMPLE RUN  ======================
```

5. Eight-bit float (15 points)

The IEEE 754 standard regarding floating point representations does not provide a specification for 8-bit floating point numbers. Imagine, however, that such numbers adhere to the following format:

```
   x     x x x     x x x x
(sign) (exponent) (mantissa)
```

That is, 1 sign bit, 3 exponent bits, and 4 bits for the mantissa.

Write a program that prompts a user to enter 8 bits and outputs the decimal equivalent. Make sure the result is printed with three digit precision (i.e., "%.3f" in printf).

Hint: The exponent bias for this 8-bit representation is 3.

```
============= START OF SAMPLE RUN ======================
Enter 8-bit floating point number:
0 0 0 0 0 0 0 0
The decimal value is 0.125
=============  END OF SAMPLE RUN  ======================


============= START OF SAMPLE RUN ======================
Enter 8-bit floating point number:
1 1 0 1 0 1 1 0
The decimal value is -5.500
=============  END OF SAMPLE RUN  ======================
```