Lab Practice Week 4
To be submitted as C Exercise 1 by Week 5

**Internal Students:** You need to show a working version of your solutions to program 8, 9 and the Number Conversion Exercise. Your tutor will expect to see your submission during your lab class in or before week 5.

**External Students:** Please email your Program 8, 9 and the Number Conversion Exercise to your tutor. Your tutor will expect to receive them by the end of week 5.

### Program 8

Write a program in C with the following functions:

- A function that reads in 10 float numbers and returns the input to the called function.
- A function for determining the smallest number of the 10 inputs.
- A function for determining the largest number of the 10 inputs.
- A function for displaying the original 10 inputs.
- A function for calculating the average of the 10 inputs.

One is to expect the main program with following layout:

```c
getInput(input);
smallestNumber = getSmallestNumber(input);
largestNumber = getLargestNumber(input);
average = calculateAverage(input);

outputOriginalNumbers(input);

Display the average of the inputs.
Display the smallest number.
Display the largest number.
```

Question: Functions such as getSmallestNumber, getLargestNumber and calculateAverage are not meant to alter the value of the input. What can one do to ensure that? Or more specifically, what should be added in the parameter declaration to ensure this?
Program 9

Write a program in C that will read from the keyboard a binary number (base-2 form) and display in a non-negative integer in the range 0 to 32767 (base-10 form). Assume that the size of the integer is 16 bits.

For example,
User input: 0000000000011011
Program output: 27
User input: 000001000101011
Program output: 555

Number Conversion Exercise

Perform the following exercises by hand. Do not use a calculator to perform the number conversions. Show full workings of your solutions.

1) Convert the following unsigned binary number to
   a. an unsigned decimal number, and
   b. to an unsigned hexadecimal number:
      10101111

2) Convert the following unsigned hexadecimal number to
   a. an unsigned decimal number and,
   b. to a binary number:
      CC7

3) Convert the following decimal number to a hexadecimal number, and then from hexadecimal number to its equivalent in octal:
   2102

4) Using 8-bit representation, convert the following decimal number to its binary equivalent in two's complement representation:
   -72

Note: While you may develop your program in any platform or compiler, all programs are expected to be able to run in the Cygwin environment. Even you may not be able to produce the complete program in the lab session, you MUST record and show your work to your tutor, otherwise no marks will be given. You should be able to give the following information:

1. An overall design of the program (How does the program work?)
2. Algorithm of the program (How do you process the data?)
3. Code and comments (What have you developed?)
4. Results (What are the outputs from the program?)
5. Testing (How did you test it?)
6. Discussion (Does it work? If no, what will you do next? If yes, how can it be improved?)