Data Structures and Program Design Using C

SCS 1301

AY 22 - Semester 1

Exercise IV

Dr Manjusri Ishwara

1. Write C programs to produce the following formatted inputs given below using the *scanf*(...) function.
   (a) Read user input to five variables 'a', 'b', 'c', 'd' and 'e' as specified below and output the same using the *printf*(...) in the given format and the integer format.
      i. an integer to variable 'a'
      ii. a floating point to variable 'b'
      iii. a character to variable 'c'
      iv. an octal number to variable 'd'
      v. a hexadecimal number to variable 'e'
   (b) Write a program that takes an integer as input from the user and prints it in the standard output stream or provides an error message if the user inserts non-integers.
   (c) Convert parts (a) and (b) of question 2 in Exercise III so the user can input the relevant inputs.
   (d) Extends the above program so it continues until the user inputs a negative number.
2. The following exercises are to be completed using **pointers**.
   (a) Add two numbers using pointers and save the result in the memory location of one of the two numbers.
   (b) What is the size of a pointer?
   (c) Declare a pointer to an int, double, and char in that order and perform five(5) increment operations using a loop on the three-pointers. Explain the behavior.
3. The following exercises are to be completed using **functions** where necessary.
   (a) Write a program to calculate and return the power of a base number. Both the power and the base number are to be provided by the user. For example, if the user inputs three (3) as the number and four (4) as the power, the returned value should be 81.
   (b) Write a function to return whether a given number is odd or even.
   (c) The Greatest Common Divisor (GCD) is a number that is the largest common term for a given set of numbers.
      i. Write a function to take two numbers and compute the GCD by finding the greatest division by starting from one (1) and looping to the maximum number.

ii. Write another function that takes two numbers and computes the GCD using the least common multiplier method (LCM). The computation of LCM should be a separate function. Use the following equation to compute the GCD using LCM.

$$GCD(a,b) = \frac{a \times b}{LCM(a,b)}$$

iii. Extend the above program to compute the GCD for any number of numbers.

(d) Write a function to take any trigonometric function, the starting angle and ending angles and output a trigonometric table from the starting angle to the ending angle with its value in radians with a thirty (30^o) degrees step.

i. Extend the above function to have an arbitrary step size supplied to the function as a parameter.

ii. Extend the program to accept the relevant parameters via user input.

iii. Extend the program to run infinitely until the user enters 'X' to end the program.

(e) Write a C program that outputs the command line menu as given below.

```
##########################################
##   SCS 1301 - Scientific Calculator   ##
##                by                    ##
##             <your name>              ##
##########################################
##########################################
#   1.  Addition        8.  Sine         #
#   2.  Subtraction     9.  Cosine       #
#   3.  Multiplication  10. Tangent      #
#   4.  Division        11. Inverse      #
#   5.  Power           12. Remainder    #
#   6.  LCM             13. Odd or Even  #
#   7.  GCD             14. Percentage   #
#                                        #
#   15. Exit                             #
##########################################
##########################################
```

i. Implement each option given in the menu as a separate function. Choose the appropriate return type for each function. For each implementation, only the two operand cases are to be considered.

ii. Extend the functionality so that when the user enters the relevant menu item number, the relevant function is called. The two operands for the function are to be obtained from the user after selecting the relevant function.

iii. Extend the program's functionality so it performs computations until the user exits by inserting 15.

iv. Extend the program to have option 16, which prints the history of the operations performed with the relevant operands.

v. Extend the program functionality by giving a sub-menu to the division option as given below. Write appropriate functions to implement the functionality.

```
##########################
##########################
##  1. Integer Division ##
##  2. Normal Division  ##
##########################
##########################
```

vi. Extend the program functionality when integer operations are performed. The output should be given as integers without the decimal point and vice versa.

(f) Write a program that contains a function that outputs "True" when the number entered as a parameter is a floating point number.

(g) Write a program that contains a function that accepts a single character as an argument and returns the opposite case. If a' was inserted, the output should be 'A' and vice versa.

(h) Write a program that contains a function that calculates whether a given data point in X and Y coordinates are on a circle with the origin O_x and O_Y and radius R and returns "True".

i. Extend the functionality of the above function to compute the following and display the output in the standard output stream.

- Circumference of the Circle
- Area of the Circle
- Volume of a sphere of the same radius.

(i) Write a program that takes two XY coordinate points and returns the angle the line creates with the X-axis in radians.

(j) Consider four(4) XY coordinate points, with the first two belonging to line A and the latter two belonging to line B. Write a program to output the intersection point of the lines A and B. If there is no intersection, an appropriate message has to be provided.

4. Complete the following exercises using the **array** data structures.

(a) Write a program to create an integer array of size ten (10) and insert values provided by the user.

(b) Write a program that creates an array of eight (8) the user provides and allocates the given number of bits. The user is then asked to store 1s or 0s for each element in the array. Convert the stored number into decimals and display it.

(c) Write a program that will store an arbitrary number of characters in an array and identify whether it is a palindrome or not. E.g., RACECAR, WOW, TACO CAT

(d) Consider an array of ten (10) integers. Pointers are used to access each element of the array and print it in the standard output.

(e) Consider an array of 100 integers that are initialised at random. Write four (4) functions to find the average, minimum, maximum and standard deviation of the values in the array. The array has to be passed to each function as a parameter, and the result has to be returned to the caller.

(f) What would be the output for each print statement in the code given below? You must dry run first and then execute the code to verify the results.

```
int main()
{
    int arr[10];
    arr[1] = 97;
    arr[2] = 75;
    arr[7] = 50;
    arr[9] = 3;
    arr[0] = arr[1] * arr[9] - arr[2];
    arr[3] = 5 * arr[7];
    arr[4] = 1;
    arr[8] = arr[4] + arr[4];

    printf("%d\n",arr[9]);

    for(short i = 9; i >= 0; i--)
    {
        printf("%d = %d\n",i, arr[i]);
    }
    return 0;
}
```

(g) What would be the output for each print statement in the code given below? You must dry run first and then execute the code to verify the results.

```
int main()
{
    char welcome[] = {'H', 'E', 'L', 'L', 'O'};

    printf("%c\n", welcome[3]);

    welcome[0] = 'Y';

    printf("%c\n", welcome[3] + 2);

    for(unsigned short j = 0; j < 3; j++)
    {
        printf("%c\n",welcome[j]);
    }
    return 0;
}
```

*****