CS F111 - Computer Programming - Lab 1

Date: March 23, 2021 - 5pm to 7pm.

- The lab is **EVALUATIVE**.
- Follow the instructions given below in the exact order.
- Any deviation from the instructions or incomplete steps will be dealt with according to the policy announced on quanta.
- Without the video recording link, the lab marks will be withheld.
- You may refer **ONLY** to the teaching materials shared by the course instructors.

LAB INSTRUCTIONS

(Please ensure that you follow the instructions in this order.)

1. Close all applications and browser-tabs except the ones needed during the lab, and join the Google meet assigned to your group..
2. Start recording your screen and webcam feed in the format mentioned in the "Software Prerequisites" document. Ensure that the date/time are visible.
3. Solve the questions given in the question paper.
4. When you are ready to submit your solution, upload your C program via the form given below:
   https://forms.gle/TDc5Q9dcMmuKVrTZ8
   Please ensure that you are using BITS email ID while filling the form.
5. Stop screen and webcam recording.
   Please click the "Stop recording" button only once. If you click it multiple times, you may lose the entire recording.
6. Upload the recording on your BITS Google Drive.
7. Edit the options on the uploaded recording to allow the "All can view" option and copy the link to be shared. If you're unsure about this, use the following link : https://tinyurl.com/GDriveuploadhelp
8. Submit the link of the recording via the form below by 5pm, 24 March:
   https://forms.gle/avc6PvKCT18F5ZHC8
   Please ensure that you are using BITS email ID while filling the form.

QUESTIONS

Question 1 - (2 Marks):

Note the compiler warnings in each of the cases below and save them in a file named **exl.txt** . You may use any text editor of your choice to do this.

a) If there is a mismatch between the number of type identifiers and the variables, then the compiler shows a warning.

- ```
  int a,b;
  printf("%d,%d",a);
  ```

- ```
  int a,b;
  printf(“%d”,a,b);
  ```

b)If there is a mismatch between the kind of type identifier and the variable, then the compiler shows a warning.

```
int a,b;
printf(“%d,%f”,a,b);
```

Question 2 - (2 Marks):

- Integer datatype **`int`** can be prefixed with the keyword **`short, long, unsigned`** and **`signed`** in order to change the range of values.
  For example : **`short int x;`**
- Similarly **`char`** can be prefixed with keywords **`signed`** and **`unsigned`**.
- Sometimes a keyword alone refers to some default datatype.
  For example : **`short x;`**

Identify the size and format specifier of each data type given below and save the results in a file named **ex2.txt.**

Datatype	Format Specifier	Bytes
`signed char`		
`unsigned char`		
`int`		
`unsigned int`		
`short`		
`unsigned`		
`long`		
`unsigned long`		

HINT :

- For identifying the size of a datatype, use the **`sizeof()`** function.
- For identifying the format specifier of a datatype, declare a variable of that datatype and try to print the contents of the variable using any format specifier. If your format specifier is incorrect, then the compiler gives a warning along with the correct format specifier to be used.

Question 3 - (2 Marks):

Write a C program titled **ex3.c** that does the following:

1. Print the integer 4294967293 by modifying only the format specifier in the code given below.

```
printf("%d\n",4294967293);
```

2. Print the following output by modifying only the format specifier in the code given below.

```
    1
   111
  11111
 1111111
111111111
```

```
printf("%9d\n",1);
printf("%9d\n",111);
printf("%9d\n",11111);
printf("%9d\n",1111111);
printf("%9d\n",111111111);
```

Question 4 - (2 Marks):

Write a C program titled **ex4.c** to print the below text using a single **`printf`** statement.

Martin said, “I’m going over to Jennifer’s house for a few hours.”

“You can’t be serious!” cried Fauntleroy.

“Oh, but I am,” Martin replied.

“How will you get there?” Fauntleroy asked.

Question 5 - (2 Marks):

We know that the inbuilt function **`printf`** can be used in a program by adding the header **`#include <stdio.h>.`** Similarly there is a function **`pow(x,y)`** which calculates "**`x`** to the power **`y`**", and this can be used in a program by adding the header **`#include <math.h>`**

If you want to use both these functions, then you must add both the headers as given below:

```
#include <stdio.h>
#include <math.h>
```

Write a C program titled **ex5.c** that uses **`pow(x,y)`** to calculate 5^4 and print the resulting value.