CS110: Computing Laboratory

Department of CSE, IIT Guwahati

Lab #	**04**
Session	**AL1**
Held on	**Monday, 03-Apr-2023**
Timings	**14:00 to 17:00 Hours**
Lead TA	**Mridul Jyoti Roy**
Remark	**Quiz - 1**
# Pages	**6**
# Questions	**2 Questions**
Marks	**39 Marks**
Submission time	**16:00 Hrs, 03-Apr-2023**
Instructor	**Prof. Jatindra Kumar Deka**, **Dr. V. Vijaya saradhi**

Important note:

Question 1 file naming Answer all the tasks of the first question in a file named after your roll number. That is if your roll number is: `220101999` then first question file name should be `220101999-1.c`

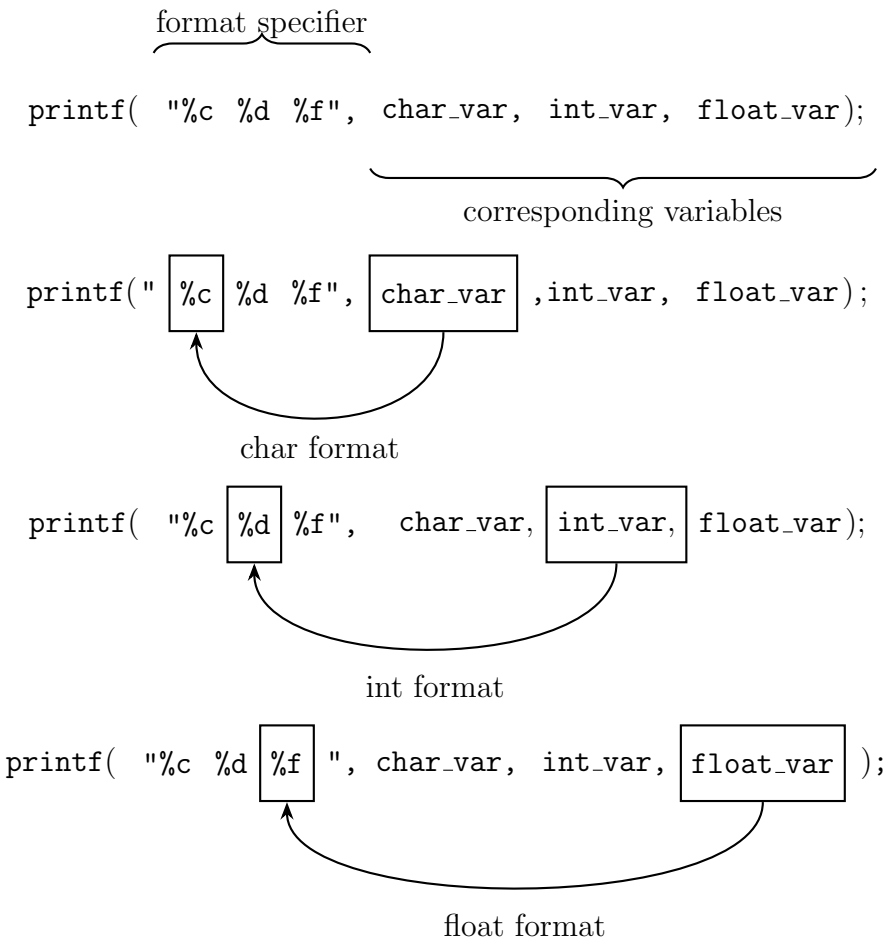
Question 2 file naming Answer all the tasks of the second question in a file named after your roll number. That is if your roll number is: `220101999` then second question file name should be `220101999-2.c`

Penalty For each incorrect file name 1 mark is deducted.

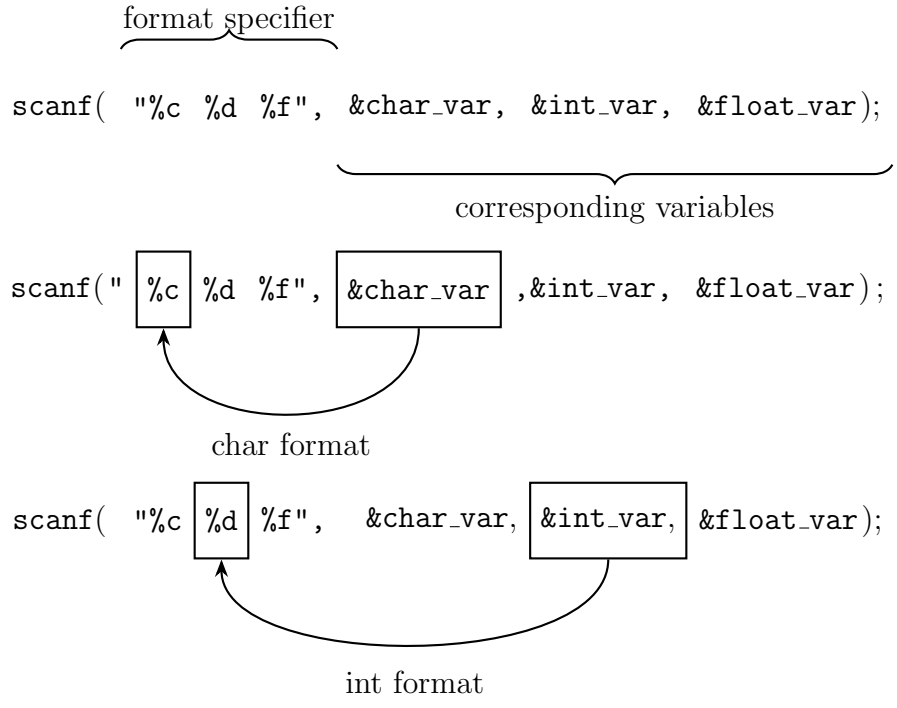
During the quiz unfair means

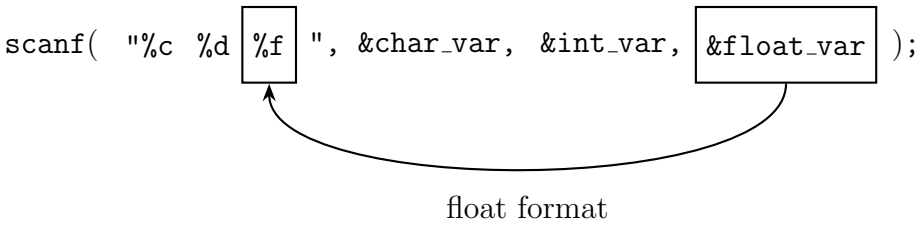
1. Discussions with fellow student is considered unfair means.
2. Peeping into the monitor of neighboring student monitor is considered unfair means.
3. Brining note books, text books or papers inside the CS110 lab is considered unfair means.
4. Brining any form of electronic device inside the CS110 lab is considered unfair means.
5. Staying in the CS110 lab after the evaluation is completed for long duation is considered unfair means.
6. Any other activity not written here but suggestive of unfair means must be avoided.

Printing format specifier



Scanning format specifier





Question 1: (13 points)
Write a C program for the following tasks.

Task 01 Data types

1. (1 mark) `(singed) character data type`
   (a) Declare one variable of data type `signed character`
   (b) Read the input from the key board into this variable
   (c) Print the value as `signed character`
   (d) Print the value as numeric value
   - Hint 1: For printing `signed characters` the format specifier is: `%c`.
   - Hint 2: For printing equivalent numerical output, the format specifier is: `%hhi`.
   - Hint 3: Refer to the diagrams given above to understand the format specifier and its role in `printf` function.
   - Hint 4: Refer to the diagrams given above to understand the format specifier and its role in `scanf` function.
2. (1 mark) `(unsigned) character data type`
   (a) Declare one variable of data type `unsigned character`
   (b) Read the input from the key board into this variable
   (c) Print the value as `unsigned character`
   (d) Print the value as numeric value
   - Hint 1: For printing `unsigned character`, the format specifier is: `%c`.
   - Hint 2: For printing equivalent numerical output, the format specifier is: `%hhu`.
3. (1 mark) `(signed) short integer data type`
   (a) Declare one variable of data type `signed short integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `signed short integer`
   (d) Hint: format specifier: `%hi`.
4. (1 mark) `(unsigned) short integer data type`
   (a) Declare one variable of data type `unsigned short integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `unsigned short integer`
   (d) Hint: format specifier: `%hu`.

5. (1 mark) `(signed) integer data type`
   (a) Declare one variable of data type `signed integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `signed integer`
   (d) Hint: format specifier: `%i` or `%d`.
6. (1 mark) `(unsigned) integer data type`
   (a) Declare one variable of data type `unsigned integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `unsigned integer`
   (d) Hint: format specifier: `%u`.
7. (1 mark) `(signed) long integer data type`
   (a) Declare one variable of data type `signed long integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `signed long integer`
   (d) Hint: format specifier: `%li` or `%ld`.
8. (1 mark) `(unsigned) long integer data type`
   (a) Declare one variable of data type `unsigned long integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `unsigned long integer`
   (d) Hint: format specifier: `%lu`.
9. (1 mark) `(signed) long long integer data type`
   (a) Declare one variable of data type `signed long long integer`
   (b) Read the input from the key board into this variable
   (c) Print the value as `signed long long integer`
   (d) Hint: format specifier: `%lli` or `%lld`.
10. (1 mark) `(unsigned) long long integer data type`
    (a) Declare one variable of data type `unsigned long long integer`
    (b) Read the input from the key board into this variable
    (c) Print the value as `unsigned long long integer`
    (d) Hint: format specifier: `%llu`.
11. (1 mark) `Real floating point data type`
    (a) Declare one variable of data type `float (real floating point)`
    (b) Read the input from the key board into this variable
    (c) Print the value as `real floating point` using all format specifiers given below
        - use format specifier: `%f`.
        - use format specifier: `%g`.
        - use format specifier: `%e`.
        - use format specifier: `%a`.

12. (1 mark) `Extended precision floating point data type`
    (a) Declare one variable of data type `double`
    (b) Read the input from the key board into this variable
    (c) Print the value as `double` using all format specifiers given below
        - use format specifier: `%lf`.
        - use format specifier: `%lg`.
        - use format specifier: `%le`.
        - use format specifier: `%la`.

Task 02 (1 mark) Compile the program of question 1. During the compilation, rename the output of the executable to `question1`

Question 2: (26 points)

Write another C program for the following tasks.

Task 03 Let $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$, $B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$ and $C = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$

1. (1 mark) Declare nine integer data types corresponding to matrix A.
2. (1 mark) Declare nine integer data types corresponding to matrix B.
3. (1 mark) Declare nine floating point data types corresponding to matrix C.
4. (1 mark) Read 9 integer data types into the variables corresponding to matrix A.
5. (1 mark) Read 9 integer data types into the variables corresponding to matrix B.
6. (1 mark) Read 9 floating point data types into the variables corresponding to matrix C.
    - (1 mark) demonstrate the capability of your program to take input via keyboard.
    - (2 marks) demonstrate the capability of your program to take input via an input file containing 18 integers and 9 floating point numbers.
    - `AL1-input-1.txt` file description: First 9 lines contains data corresponding to matrix A. Next 9 lines of data corresponds to matrix B. Following 9 lines of data corresponds to matrix C.
    - (1 mark) Refer to `AL1-output-1.txt` for example expected output.
    - (4 marks) You are given four more additional inputs for demonstrating the correctness of your program.
7. (9 marks = 3 + 3 + 3; 3 marks for correct multiplication; 3 marks for correct addition) Perform an **equivalent** of matrix multiplication and addition given as $D = (A \times B) + (B \times C)$ using all the above variables.
8. (2 marks) While evaluating the above expression, print the intermediate result as stated below:
    (a) (1 mark) Print the result of $B \times C$.

(b) (1 mark) Print the result of $A \times B$.

9. (1 mark) Print the resulting output in the matrix form given as

$$\begin{matrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{matrix}$$

Task 04 (1 mark) Compile the program of question 2. During the compilation, rename the output of the executable to `question2`