



# MASENO UNIVERSITY UNIVERSITY EXAMINATIONS 2024/2025

SECOND YEAR FIRST SEMESTER EXAMINATIONS FOR THE  
DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE  
AND BACHELOR OF EDUCATION SCIENCE WITH  
INFORMATION TECHNOLOGY

## MAIN CAMPUS

### CCS 219: SYSTEMS AND PROCEDURAL PROGRAMMING

Date: 30<sup>th</sup> January, 2025

Time: 12. 00 - 3. 00 pm

#### INSTRUCTIONS:

- DO NOT write anywhere on this question paper
- Answer all questions in SECTION A and any other TWO from SECTION B
- Fasten together all loose answer sheets used
- No mobile phone in the examination room

ISO 9001:2015 CERTIFIED



## SECTION A: ANSWER ALL QUESTIONS

### Question one (30 Marks)

a) State the functions of Process control subsystem of UNIX. (4 marks)

b) Differentiate between the following terms/phrases as used in C programming using relevant examples.

- i) A variable and a constant: - A memory location whose value does not change during program execution (2 marks)
- ii) Scanf and printf. → used to output information to standard output from std::cout (2 marks)
- iii) Function prototype and function definition. → Reads the input from the standard input stream std::cin and scan that input according to format provided (2 marks)

c) Describe the three types of accounts on a Unix system. (3 Marks)

d) Write a program in C language that will receive three integer numbers from the keyboard then determines and displays the biggest number of the three on the screen. (3 marks)

e) Write a UNIX command line statements that will do the following

i) Create a directory called project1 on project directory (1 mark)

ii) Deletes a directory called notes that contains other directories (1 mark)

f) Write a C program that will accept three floating point numbers from the keyboard, compute their sum and print the result on the screen. (3 marks)

g) State and briefly describe three ways that can be used to implement inter-process communication (IPC) synchronously. (6 Marks)

h) Write a C program that will generate integer numbers between 1 and 3000 that are divisible by 11, compute their sum then display the numbers and their sum on the screen. (3 marks)

```
#include <stdio.h>
int main () {
    float num1, num2, num3, sum;
    printf ("Enter first number:\n");
    scanf ("%f", &num1);
    printf ("Enter second number:\n");
    scanf ("%f", &num2);
    printf ("Enter third number:\n");
    scanf ("%f", &num3);
    printf ("The sum = %f\n");
    return 0;
}
```

d) int  
#include <stdio.h>  
int x, y;  
int main () {  
 printf ("In Enter value of x\n");  
 scanf ("%d", &x);  
 printf ("In Enter value of y\n");  
 scanf ("%d", &y);  
 if (x == y)  
 printf ("x equals to y\n");  
 else if (x > y)  
 printf ("x is greater than y\n");  
 else  
 printf ("x is smaller than y\n");  
 return 0;
}

## SECTION B: ANSWER ANY TWO QUESTIONS

### Question Two (20 marks)

- a) Briefly explain the following common UNIX server software. (8 marks)
- Domain Name System
  - Telnet
  - File transfer protocol
  - World Wide Web

- b) Use a five-state process model to explain the process state transition in UNIX operating system.

- i) Describe briefly any three algorithms that can be used to allocate memory for a new process or an existent process that has to be swapped in. (6 Marks)

Best fit, First fit, Function hide, Implementing details, allowing programmer to understand them, easier to debug and testing individual without affecting whole program (6 marks)

### Question Three (20 Marks)

- a) State four advantages of using functions in C programming. (4 marks)
- b) Write a C program that receives three decimal numbers from the keyboard then uses both function prototype and function definition to computes their sum and displays the result on the screen.

```
#include <stdio.h>
int main ()
{
    int choice;
    char grade;
    printf("Enter 1: marks 0 to 100\n");
    scanf("%d", &choice);
    printf("Enter marks 60 to 69\n");
    scanf("%d", &choice);
    printf("Enter marks 50 to 59\n");
    scanf("%d", &choice);
    printf("Enter 4 marks 0 to 39\n");
    scanf("%d", &choice);
}
```

switch (choice) {

case 1:

grade = 'A';

printf("The grade = %c\n", grade);

break;

case 2:

grade = 'B';

printf("The grade = %c\n", grade);

break;

case 3:

grade = 'C';

printf("The grade = %c\n", grade);

break;

case 4:

grade = 'D';

printf("The grade = %c\n", grade);

break;

case 5:

grade = 'E';

printf("The grade = %c\n", grade);

- c) What is a control structure? Write a C program that demonstrate how a switch decision construct

(6 marks)

works.

int choice; // Enter marks 60 to 69

char grade; // Enter marks 50 to 59

long sum = 0; // Enter 4 marks 0 to 39

const int upper\_limit = 3000; // Enter 1: marks 0 to 100

const divisor = 7; // Enter 2: marks 60 to 69

for (i = 1; i <= upper\_limit; i++) {

if (i % divisor == 0) {

sum += i; // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

printf("The sum of all numbers divisible by 7 is %d\n", sum);

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39

} // Enter 1: marks 0 to 100

return 0; // Enter 2: marks 60 to 69

} // Enter 3: marks 50 to 59

} // Enter 4: marks 0 to 39</

#### Question Four (20 Marks)

- c) Draw a well labelled block diagram showing the design of a Unix Operating system and briefly explain the functions of its various components. (17 Marks )
- d) Explain the three modes in which a vi text editor can operate. (3 Marks )

#### Question Five (20 Marks)

- a) Compare between an array and a structure in C programming language. (4 Marks)
- b) Declare a record called Refugee that has the following fields: identityNo, name, gender, of type string, age of type integer, native country of String, hosting country of type string and camp of type string. (3 marks)
- c) Write a program segment that declares a variable of type Refugee and assign values to each field. (3 marks)
- d) Write a C program segment that declares a single dimensional array called RefugeeRecord of type Refugee that can contain records of 10 refugees entered from the keyboard. (3 marks)
- e) Write a C program that will print the records of the ten refugees on the screen. (3 marks)
- f) String is considered as an array of characters in C programming language.
  - i) Write a program statement that will declare and initialize a string of 10 elements. (1 marks)
  - ii) Use appropriate examples to explain the use of gets() and puts() functions as applied on strings. (3 marks)