Course No. : EE207

# EE207-Computer Programming

Question bank

**Syllabus :-**

| Module | Contents | Hours | Sem Exam Marks |
|---|---|---|---|
| I | **Introduction to Programming**: Machine language, assembly language, and high level language. Compilers and assemblers. Flow chart and algorithm – Development of algorithms for simple problems.<br>**Basic elements of C**:Structure of C program –Keywords, Identifiers, data types, Operators and expressions – Input and Output functions | **5 hours** | **15%** |
| II | **Control statements in C**:if, if-else, while, do-while and for statements, switch, break, continue, go to, and labels.<br>Programming examples | **7 hours** | **15%** |
| III | **Arrays and Strings**: Declaration, initialisation, processing<br>Array and string two dimensional and multi-dimensional arrays –application of arrays. Example programs. | **7 hours** | **15%** |
| IV | **Functions** : Functions declaring, defining, and accessing<br>functions –parameter passing methods – passing arrays to functions , Recursion .<br>**Storage classes**: extern, auto, register and static.<br>Example programs. | **7 hours** | **15%** |
| V | **Structures** : declaration, definition and initialization of structures, unions<br>**Pointers**: Concepts, declaration, initialization of pointer variables, Accessing a Variable through its Pointer Chain<br>of Pointers, Pointer Expressions, Pointer Increments and Scale Factor, Pointers and Arrays, examples | **8 hours** | **20%** |
| VI | **File Management**: File operations, Input/output Operations on Files, Random Access to Files, File pointer<br>**Introduction to Python** :Basic Syntax, Operators, control statements, functions-examples | **8 hours** | **20%** |
| **Final exam** | | | |

1. Write a program to print "Hello world" on the screen
2. Write a program to read name of the user, and greet "Good Day" to user.
3. Read the year of birth of the user. Then print the year in which he celebrate his 50th birthday
4. Read two numbers a and b. Print a+b, a-b, a*b and a/b
5. Find average of 3 numbers
6. Write a program for computing bank interest (I= PNR)
7. Write a program to calculate value of resistance using ohm's law
8. Calculate BMI of a person by reading Mass and Height (in Meter)
9. Read the length in centimeter. Conver it into meters
10. Read the time in seconds. Then convert it into day: hrs: mints: seconds (eg: 100000 meas 1 day: 3 hrs: 40 mints: 40 seconds)
11. Write a program that takes the radius of a sphere (a floating-point number) as input and outputs the sphere's diameter, circumference, surface area, and volume.

12. Read number of days, hours and minutes. Then convert it into seconds
13. Convert centigrade to farenheat.
14. An employee's total weekly pay equals the hourly wage multiplied by the total number of regular hours plus any overtime pay. Overtime pay equals the total overtime hours multiplied by 1.5 times the hourly wage. Write a program that takes as inputs the hourly wage, total regular hours, and total overtime hours and displays an employee's total weekly pay.

15. Write a program to find area and perimeter of square
16. Write a program to find area and perimeter of rectangle
17. Write a program to find area and perimeter of circle
18. Compute the value of f(x)=3x+6
19. Compute the value of f(x)=x^4+3x^3+2x+7
20. Read two numbers and b, then print quotient and reminder after the division a/b
21. Interchange values of variables using a temporary variable
22. Exchange values of variables without using temporary variable
23. Write a program to read number and print the following: Print "Even" if it is even number and print its square
24. Check whether a number is odd or even
25. Read two numbers. Check whether first number is divisible by second number
26. Read a number. Print "Yes" if it is an odd number divisible by 7.
27. For attending the physical test of Army Service, the candidate must have minimum height 160 cm. Read height of candidate and state whether he is eligible or not
28. Read a phone number from user Then ask him to retype the number for confirmation. Verify that whether both numbers are same.
29. Read a user name (string) and password (number) from user. If it match with predefined user name and password, print "Welcome user" otherwise "Login Failed"
30. Read a character. Check whether it is vowel or not.
31. Read marks of a student(out of 50) in four subjects. Grade of the studnet is calculated as follows:
    1. Must secure minimum 30 marks in all subjects
    2. if total marks > 90% , S grade total marks between 80- 90% A grade, 70- 80% B grade, otherwise C grade.
    3. If he failed in any subject, the grade will be U, whatever be the total marks
30. The salary of person is basic pay- Tax deduction. For male employees salary above 15000, 2% will be the tax. For womern employees of this salry class 1% will be the tax. For employees salary below 15000, no tax deduction. Calculate net salary of an employee.
31. Read starting time and ending time of a function. Then calculate its duration.

32. Write a program to implement NAND gate, NOR and XOR (read values of x and y, produce the corresponding result)
33. Write a program to print numbers from 1 to N
34. Write a program to print sequence 1,3,5,7,...
35. Write a program to print the sequence 1,4,9, 16,....
36. Write a program to print the sequence 0,1,1,2,5,27,734,....
37. Print the sequnce 1,2,2,4,8,32,256,....
38. Read n numbers and find their sum
39. Average of N numebrs
40. Print the pattern:

```
1 2 3
2 3 4
3 4 5
4 5 6
```

41. Print the pattern:

```
1
2 3
4 5 6
7 8 9 10
```

42. Print the pattern:

```
1
1 2
1 2 3
1 2 3 4
```

43. Print the pattern:

```
      1
    2   3
  4   5   6
6   7   8   9
```

44. Print multiplication table of N (upto 10)
45. Print factorial of N
46. Print factors of a numbers
47. Check whether a number is prime or not
48. Write a program to find the sum of digits of a number
49. Read two numbers a and b, print the sum of numbers between a nd b, which are divisible by 7
50. Count the number of numbers divisible by both 6 and 8, between limits a and b

## Some examples

### Sum of digits

```
#include <stdio.h>
int main()
{
        int n, sum = 0, remainder;
        printf("Enter an integer\n");
        scanf("%d", &n);
        while (n != 0)
        { remainder =n%10;
          sum = sum + remainder;
          n = n / 10;
        }
        printf("Sum of digits of entered number = %d\n", sum);
        return 0;
}
```

OUTPUT
Enter an integer 1659
Sum of digits of entered number = 21

### Add digits using recursion

```
#include <stdio.h>
int add_digits(int);
int main()
{
        int n, result;
        printf("Enter a number\n");
        scanf("%d", &n);
        result = add_digits(n);
        printf("%d\n", result);
        return 0;
}
        int add_digits(int n) {
        static int sum = 0;
        if (n == 0)
        { return 0; }
        sum = n % 10 + add_digits(n / 10);
        return sum;
        }
```

OUTPUT
Enter a number 342
Ans 9

**Add 2 numbers using pointers**

```
#include <stdio.h>
int main()
{
        int first, second, *p, *q, sum;
        printf("Enter two integers to add\n");
        scanf("%d%d", &first, &second);
        p = &first;
        q = &second;
        sum = *p + *q;
        printf("Sum of entered numbers = %d\n",sum);
        return 0;
}
```

OUTPUT
Enter two integers to add 33, 4
Sum of entered numbers = 37

**Addition of 2 matrix**

```
#include <stdio.h>
int main()
{
        int m, n, c, d, first[10][10], second[10][10], sum[10][10];
        printf("Enter the number of rows and columns of matrix\n");
        scanf("%d%d", &m, &n);
        printf("Enter the elements of first matrix\n");
        for (c = 0; c < m; c++)
        {
                for (d = 0; d < n; d++)
                {
                  scanf("%d", &first[c][d]);
                }
        }
        printf("Enter the elements of second matrix\n");
        for (c = 0; c < m; c++)
        {
```

```
            for (d = 0 ; d < n; d++)
            {
              scanf("%d", &second[c][d]);
             }
          }
          printf("Sum of entered matrices:-\n");
          for (c = 0; c < m; c++)
                  {
                   for (d = 0 ; d < n; d++)
                  {
                    sum[c][d] = first[c][d] + second[c][d];
                    printf("%d\t", sum[c][d]);
                  }
                  printf("\n");
                  }
          return 0;
}
```

OUTPUT
Enter the number of rows and columns of matrix 2 2
Enter the elements of first matrix 1 2 3 4 Enter the elements of second matrix
1 2 3 4
Sum of entered matrices:- 2 4 6 8

**Check prime no. or not**

```
#include <stdio.h>
main()
{       int n, c = 2;
        printf("Enter a number to check if it is prime\n");
        scanf("%d", &n);
        for (c = 2; c <= n - 1; c++)
        {
                if (n % c == 0)
                { printf("%d is not prime.\n", n);
                break;
                }
        } if (c == n)
          printf("%d is prime.\n", n);
          return 0;
}
```

OUTPUT
Enter a number to check if it is prime 99
99 is not prime.

**File copying**

```
//at first we have to create file1.txt in the same directory and write something
in that file1.txt
so that it will be copied to the other file*/
// here file1.txt is a already created and defined file*/



#include<stdio.h>
int main()
{
        FILE *filepointer1,*filepointer2;
        char a;
        filepointer1=fopen("file1.txt","r");
        filepointer2=fopen("file2.txt","w");
        do
                {
                a=fgetc(filepointer1);
                fputc(a,filepointer2);
                }
                while(a!=EOF);
        if (filepointer2)
                {
                while (fscanf(file2, "%s", a)!=EOF)
                printf("%s",a);
                fclose(filepointer1);
                fclose(filepointer2);
                }
        Printf("file contents are successfully copied");
        return(0);
}
```

OUTPUT
file contents are successfully copied

**Fibonacci series**

```
#include <stdio.h>
void main()
{
        int a, b, c, i, n;
        a = 0;
        b = 1;
```

```
        printf("Enter a number to define the length of fibonacci series: ");
        scanf("%d", &n);
        printf("\nThe Series is: \n");
        printf("%d\t%d", a, b);
        for (i = 0; i < n; i++)
                {
                c = a + b;
                a = b;
                b = c;
                printf("\t%d", c);
                }
        getch();
}
```

OUTPUT
Enter a number to define the length of fibonacci series: 5
The Series is: 0 1 1 2 3 5 8

**Odd or even**

```
#include <stdio.h>
main()
{
        int n;
        printf("Enter an integer\n");
        scanf("%d", &n);
        if ((n / 2) * 2 == n)
        printf("Even\n");
        else
        printf("Odd\n");
        return 0
}
```

OUTPUT
Enter an integer 7
The number is Odd

**Leep year**

```
#include <stdio.h>
int main()
{
        int year;
        printf("Enter a year to check if it is a leap year\n");
```

```
        scanf("%d", &year);
        if ( year%400 == 0)
                printf("%d is a leap year.\n", year);
        else if ( year%100 == 0)
                printf("%d is not a leap year.\n", year);
        else if ( year%4 == 0 )
                printf("%d is a leap year.\n", year);
        else
                printf("%d is not a leap year.\n", year);
        return 0;
}
```

OUTPUT
Enter a year to check if it is a leap year 2022
2022 is not a leap year