

# C - Programming Language

## Week 5

Q1 WAP to print the following pattern

(i) #include <stdio.h>

```
int main()
{
    int n;
    printf (" Enter the value of n : \n");
    scanf ("%d", &n);
    for (int i=1 ; i<=n ; i++)
    {
        for (int j=1 ; j<=n ; j++)
        {
            printf ("%d", j);
        }
        printf ("\n");
    }
    return 0;
}
```

Name : Ayush sharma

Section : AL1

Roll.no : 14

University roll no. : 2315000569

Output

Enter the value of n :- 4

1 2 3 4

1 2 3 4

1 2 3 4

1 2 3 4

```

c) #include < stdio.h >
int main()
{
    int n;
    printf("Enter the value of n:");
    scanf("%d", &n);
    for (int i=0; i<n; i++)
    {
        for (int j=0; j<=n; j++)
            printf("*");
        printf("\n");
    }
    return 0;
}

```

### Output

Enter the value of n: 5

```

* * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

```
iii) #include < stdio.h>
int main ()
{
    int n;
    printf ("Enter the value of n: ");
    scanf ("%d", &n);
    for (int i=1 ; i<=n ; i++)
    {
        for (int j=1 ; j<=i ; j++)
        {
            printf ("%d", j);
        }
        printf ("\n");
    }
    return 0;
}
```

output

Enter the value of n: 4

1  
1 2  
1 2 3  
1 2 3 4

(iv) #include < stdio.h >  
int main()

{

int n;

printf("Enter the value of n : ");

scanf("%d", &amp;n);

for(int i=1; i&lt;=n; i++)

{

for(int j=1; j&lt;=i; j++)

{

printf("%d", j);

{

printf(" ") ;

printf("\n");

{

return 0;

}

Output

---

Output with the code

Enter the value of n : 4

1

2 2

3 3 3

4 4 4 4

```

v) #include < stdio.h >
int main()
{
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= i; j++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

Output

---

Enter the value of n: 4

\*  
\* \*  
\* \* \*  
\* \* \* \*

```

(Vi) #include <stdio.h>
int main()
{
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);
    for (int i='A'; i<='D'; i++)
        for (int j='A'; j<=i; j++)
            printf("%c", j);
    printf("\n");
    return 0;
}

```

---

### Output

Enter the value of n: - D

A

A B

A B C

A B C D

(vii) #include < stdio.h>

```
int main()
{
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);
    for (int i=1; i<=n; i++)
    {
        for (int j=1; j<=i; j++)
        {
            printf("%d", c);
            c++;
        }
        printf("\n");
    }
    return 0;
}
```

Output:

1  
2 3  
4 5 6  
7 8 9 10

```

(viii) #include <stdio.h>
int main()
{
    int n;
    printf("Enter the value of n");
    scanf("%d", &n);
    for (int i=1; i<=n; i++)
    {
        for (int j=1; j<=i; j++)
        {
            printf("%d", j%2);
        }
        printf("\n");
    }
    return 0;
}

```

Output:

Enter the value of n :- 5

```

1
10.
101
1010
10101

```

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

```
int main()
```

```
{
```

```
int n;
```

```
printf("Enter the value of n: ");
```

```
scanf("%d", &n);
```

```
for (int i = 5 ; i >= n ; i--)
```

```
{
```

```
for (int j = 5 ; j >= i ; j--)
```

```
{
```

```
printf("%d", j);
```

```
}
```

```
{
```

```
printf("\n");
```

```
}
```

```
}
```

```
return 0;
```

```
}
```

---

Output

enter the value of n: 2

5

5 4

5 4 3

5 4 3 2

5 4 3 2 1

```

(X) #include <stdio.h>
int main()
{
    int n;
    printf("Enter the value of n");
    scanf("%d", &n);
    for (int i = 1; i <= 5; i++)
    {
        for (int j = 5; j >= 1; j--)
        {
            printf("%d", j);
        }
        printf("\n");
    }
    return 0;
}

```

### Output

Enter the value of n:- 5-

5 4 3 2 1

5 4 3 2

5 4 3

5 4

5

```

⑩ #include <stdio.h>
int main()
{
    int n;
    printf("Enter the value of n:-");
    scanf("%d", &n);
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
            if (i > 0 && i < n - 1 && j > 0 && j < n - 1)
            {
                printf(" ");
            }
            else
            {
                printf("*");
            }
        }
        printf("\n");
    }
    return 0;
}

```

Output  
Enter the value of n:- 5

*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*

(X) #include < stdion.h>

```
int main ()
```

```
{
```

```
int rows;
```

```
printf ("Enter the value of rows: ");
```

```
scanf ("%d", &rows);
```

```
for (int i=0; i<rows; i++)
```

```
{
```

```
for (int j=0; j<2 * (rows-1)-i; j++)
```

```
{
```

```
printf (" ");
```

```
}
```

```
for (int k=0; k<2 * (i+1); k++)
```

```
{
```

```
printf ("*");
```

```
}
```

```
return 0;
```

```
3
```

Output

Enter the value of rows = 5

```
*  
* * *  
* * * * *  
* * * * * * *  
* * * * * * * *
```

## HOTS

(xi)

```
# include<stdio.h>
int main()
{
    int n=5;
    for (i=0; i<2n-1; i++)
    {
        int comp;
        if (i<n)
        {
            Comp = 2n-i-1;
        }
        else
        {
            Comp = 2i-n+1+1;
        }
        for (int j=0; j<Comp; j++)
        {
            printf(" ");
        }
        for (int k=0; k<2n-Comp; k++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

out put

*	*	*				
*	*	*	*	*	*	
*	*	*	*	*	*	*
*	*	*	*	*	*	*
*	*	*	*	*	*	*
*	*	*	*	*	*	*

---

## Week -6

### Programming Questions

Q. 1 Write a menu driven program to insert and delete elements of kth position to an array of size N.

Q. 2 Write the program to print the biggest and smallest element in an array.

Answer :-

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

```
int main() {
    int arr[100];
    int n, i, largest, smallest;

    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);

    printf("Enter the elements of the array: ");
    for (i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    largest = arr[0];
    smallest = arr[0];
```

---

```
for (i = 1; i < n; i++) {  
    if (arr[i] > largest) {  
        largest = arr[i];  
    }  
    if (arr[i] < smallest) {  
        smallest = arr[i];  
    }  
}  
  
printf("The largest element in the array is: %d\n", largest);  
printf("The smallest element in the array is: %d\n", smallest);  
  
return 0;  
}
```

Q. 3 Write the program to print the sum and average of an array.

Answer :-

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

```
int main() {  
    int arr[10];  
    int sum = 0;  
    float avg;  
  
    printf("Enter the elements of the array: ");
```

```
for (int i = 0; i < 10; i++) {  
    scanf("%d", &arr[i]);  
    sum += arr[i];  
}  
  
avg = sum / 10.0;  
  
printf("The sum of the elements of the array is: %d\n", sum);  
printf("The average of the elements of the array is: %f\n", avg);  
  
return 0;  
}
```

Q. 4 Write the program to sort an array using bubble sort.

Answer:-

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

```
void bubble_sort(int arr[], int n) {  
    int i, j;  
    for (i = 0; i < n - 1; i++) {  
        for (j = 0; j < n - i - 1; j++) {  
            if (arr[j] > arr[j + 1]) {  
                int temp = arr[j];  
                arr[j] = arr[j + 1];  
                arr[j + 1] = temp;  
            }  
        }  
    }  
}
```

```
    }
}
}

int main() {
    int arr[] = {5, 2, 8, 7, 1};
    int n = sizeof(arr) / sizeof(arr[0]);

    bubble_sort(arr, n);

    printf("Sorted array: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");

    return 0;
}
```

Q. 5 Write the program to search an element using linear search as well as binary search.

Answer :-

```
#include <stdio.h>

int linearSearch(int arr[], int n, int x)
```

```
{  
    for (int i = 0; i < n; i++)  
    {  
        if (arr[i] == x)  
        {  
            return i;  
        }  
    }  
    return -1;  
}  
  
int binarySearch(int arr[], int low, int high, int x)  
{  
    if (low <= high)  
    {  
        int mid = (low + high) / 2;  
  
        // If x is greater, ignore left half  
        if (arr[mid] < x)  
        {  
            return binarySearch(arr, mid + 1, high, x);  
        }  
  
        else if (arr[mid] > x)  
    }
```

```
{  
    return binarySearch(arr, low, mid - 1, x);  
}  
  
else  
{  
    return mid;  
}  
  
}  
  
return -1;  
}  
  
int main()  
{  
    int arr[] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19};  
    int n = sizeof(arr) / sizeof(arr[0]);  
    int x = 5;  
  
  
    int result = linearSearch(arr, n, x);  
    if (result == -1)  
    {  
        printf("Element is not present in array\n");  
    }  
}
```

```
    }
else
{
    printf("Element is present at index %d\n", result);
}

result = binarySearch(arr, 0, n - 1, x);
if (result == -1)
{
    printf("Element is not present in array\n");
}
else
{
    printf("Element is present at index %d\n", result);
}

return 0;
}
```

Q. 6 Take an array of 20 integer inputs from user and print the following:

- a. number of positive numbers
- b. number of negative numbers
- c. number of odd numbers
- d. number of even numbers
- e. number of 0.

---

Q. 7 Take an array of 10 elements. Split it into middle and store the elements in two different arrays. E.g.- INITIAL array:

58, 24, 13, 15, 63, 9, 8, 81, 1, 78 After  
splitting:

58, 24, 13, 15, 63

9, 8, 81, 1, 78

Answer :-

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

```
int main() {  
    int arr[] = {58, 24, 13, 15, 63, 9, 81, 88, 1, 78};  
    int n = sizeof(arr) / sizeof(arr[0]);  
    int mid = n / 2;  
  
    int arr1[mid];  
    int arr2[n - mid];  
  
    for (int i = 0; i < mid; i++) {  
        arr1[i] = arr[i];  
    }  
  
    for (int i = mid; i < n; i++) {  
        arr2[i - mid] = arr[i];  
    }  
}
```

---

---

```
printf("The first array is: ");
for (int i = 0; i < mid; i++) {
    printf("%d ", arr1[i]);
}
printf("\n");

printf("The second array is: ");
for (int i = 0; i < n - mid; i++) {
    printf("%d ", arr2[i]);
}
printf("\n");

return 0;
}
```

Q. 8 Write the program to count frequency of each element in an array.

Answer :-

```
#include <stdio.h>

int main() {
    int arr[] = {1, 2, 3, 4, 5, 1, 2, 3};
```

---

---

```
int n = sizeof(arr) / sizeof(arr[0]);
int freq[100] = {0};

for (int i = 0; i < n; i++) {
    freq[arr[i]]++;
}

for (int i = 0; i < 100; i++) {
    if (freq[i] != 0) {
        printf("%d occurs %d times\n", i, freq[i]);
    }
}

return 0;
}
```

---

## Week -7

# C Programming Questions

Q. 1 Write the program to print row major and column major matrix.

Answer:-

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

```
int main() {
    int m, n;
    printf("Enter the number of rows and columns of the matrix: ");
    scanf("%d %d", &m, &n);

    int matrix[m][n];
    printf("Enter the elements of the matrix: ");
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    printf("The matrix in row major order is: ");
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            printf("%d ", matrix[i][j]);
        }
    }
    printf("\n");
```



---

```
printf("The matrix in column major order is: ");
for (int j = 0; j < n; j++) {
    for (int i = 0; i < m; i++) {
        printf("%d ", matrix[i][j]);
    }
}
printf("\n");

return 0;
}
```

Q. 2 Write the program to print sum of a whole matrix.

Answer :-

```
#include <stdio.h>

int main()
{
    int i, j, rows, columns, sum = 0;
    printf("Enter the number of rows and columns of the matrix: ");
    scanf("%d %d", &rows, &columns);
    int matrix[rows][columns];
    printf("Enter the elements of the matrix: ");
    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < columns; j++)
        {
            scanf("%d", &matrix[i][j]);
        }
    }
}
```

```
for (i = 0; i < rows; i++)  
{  
    for (j = 0; j < columns; j++)  
    {  
        sum = sum + matrix[i][j];  
    }  
}  
printf("The sum of all the elements of the matrix is: %d", sum);  
return 0;  
}
```

Q. 3 Write a program to add and multiply two 3x3 matrices. You can use 2D array to create a matrix.

Answer :- #include <stdio.h>

```
void main() {  
    int arr1[3][3], i, j;  
    printf("\n\nRead a 2D array of size 3x3 and print the matrix :\n");  
    printf("-----\n");  
  
    printf("Input elements in the matrix :\n");  
    for (i = 0; i < 3; i++) {  
        for (j = 0; j < 3; j++) {  
            printf("element - [%d],[%d] : ", i, j);  
            scanf("%d", &arr1[i][j]);  
        }  
    }  
  
    printf("\nThe matrix is :\n");
```

```
for (i = 0; i < 3; i++) {  
    printf("\n");  
    for (j = 0; j < 3; j++)  
        printf("%d\t", arr1[i][j]);  
    }  
    printf("\n\n");  
}
```

Q. 4 Write the program to print sum of all diagonal elements, upper triangular matrix and lower triangular matrix.

Answer:-

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

```
int main() {  
    int i, j, n;  
    printf("Enter the order of the matrix: ");  
    scanf("%d", &n);  
  
    int matrix[n][n];  
    printf("Enter the elements of the matrix:\n");  
    for (i = 0; i < n; i++) {  
        for (j = 0; j < n; j++) {  
            scanf("%d", &matrix[i][j]);  
        }  
    }  
  
    printf("Diagonal elements:\n");  
    for (i = 0; i < n; i++) {
```

```
    printf("%d ", matrix[i][i]);
}
printf("\n");

printf("Upper triangular matrix:\n");
for (i = 0; i < n; i++) {
    for (j = 0; j <= i; j++) {
        printf("%d ", matrix[i][j]);
    }
    printf("\n");
}

printf("Lower triangular matrix:\n");
for (i = 0; i < n; i++) {
    for (j = i; j < n; j++) {
        printf("%d ", matrix[i][j]);
    }
    printf("\n");
}
return 0;
}
```

Q. 5 Write the program to find the frequency of odd and even elements in matrix.

Answer:-

```
#include <stdio.h>

void main()
{
```

```
static int array[10][10];
int i, j, m, n, even = 0, odd = 0;

printf("Enter the order of the matrix \n");
scanf("%d %d", &m, &n);

printf("Enter the coefficients of matrix \n");
for (i = 0; i < m; ++i)
{
    for (j = 0; j < n; ++j)
    {
        scanf("%d", &array[i][j]);
        if ((array[i][j] % 2) == 0)
        {
            ++even;
        }
        else
            ++odd;
    }
}

printf("The given matrix is \n");
for (i = 0; i < m; ++i)
{
    for (j = 0; j < n; ++j)
```

```
    printf("%d", array[i][j]);
}
printf("\n");
}

printf("\n The frequency of occurrence of odd number = %d \n", odd);
printf("The frequency of occurrence of even number = %d\n", even);
}
```

Q. 6 Write the program to find sum of each row and sum of each column of matrix.

Answer:-

```
#include <stdio.h>
int main()
{
    printf( " Enter the required number of rows and columns: \n " );
    scanf( "%d %d" , &m, &n );

    int arr[m][n];
    printf( "Enter the elements of the given matrix: \n" ) ;
    for( int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
        {
            scanf( "%d", &arr[i][j] );
        }
    }

    printf( "\nThe elements in the matrix are \n" );
    for( int i=0; i<m; i++)
    {
```

---

```
for(int j=0; j<n; j++)
{
    printf( "%d ", arr[i][j] );
}
printf( "\n" );
printf( "\n Calculation of Row Sum \n" );
for(int i=0; i<m; i++)
{
    int rowsum=0;
    for(int j=0; j<n; j++)
    {
        rowsum = rowsum + arr[i][j] ;
    }
    printf( " \n Sum of the elements in row %d is %d \n ", i, rowsum);
}

printf( "\n Calculation of Column Sum \n" );
for( int i=0; i<m; i++)
{
    int colsum=0;
    for( int j=0; j<n; j++ )
    {
        colsum = colsum+arr[j][i] ;
    }
    printf( " \n The sum of all the elements in column %d is %d\n ", i, colsum );
}

return 0;
}
```

---

Q. 7 Initialize a 2D array of 3\*3 matrix. E.g.-

1	2	3
2	3	4
3	4	5

Q. 8 A square matrix, one having the same number of rows and columns, is called a diagonal matrix if it's only non-zero elements are on the diagonal from upper left to lower right. It is called upper triangular matrix if all elements bellow the diagonal are zeroes, and lower triangular matrix, if all the elements above the diagonal are zeroes. Write a program that reads a matrix and determines if it is one of the above mentioned three special matrices.

Q. 9 Write the program to check whether the matrix is sparse matrix or not.

Answer :-

```
#include <stdio.h>

int main() {
    int rows, cols;
    printf("Enter the number of rows and columns of the matrix: ");
    scanf("%d %d", &rows, &cols);

    int matrix[rows][cols];
    printf("Enter the elements of the matrix: ");
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
```

```
    scanf("%d", &matrix[i][j]);  
}  
}  
  
int non_zero_elements = 0;  
  
for (int i = 0; i < rows; i++) {  
    for (int j = 0; j < cols; j++) {  
        if (matrix[i][j] != 0) {  
            non_zero_elements++;  
        }  
    }  
}  
  
int total_elements = rows * cols;  
  
float sparsity = (float)non_zero_elements / (float)total_elements;  
  
  
if (sparsity >= 0.5) {  
    printf("The matrix is a sparse matrix.\n");  
} else {  
    printf("The matrix is not a sparse matrix.\n");  
}  
  
return 0;  
}
```

---

## Week 8

### Programming Questions

Q. 1 Write a C program to create, initialize and use pointers.

Answer :-

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

```
int main() {
```

```
    int *ptr;
```

```
    // Initialize the pointer to the address of an integer variable
```

```
    int num = 10;
```

```
    ptr = &num;
```

```
    // Print the value of the integer variable
```

```
    printf("The value of num is %d\n", num);
```

```
    // Print the address of the integer variable
```

```
    printf("The address of num is %p\n", &num);
```

```
    // Print the value of the pointer
```

```
    printf("The value of ptr is %p\n", ptr);
```

---

```
    printf("The value of num through ptr is %d\n", *ptr);
```

```
    return 0;
```

```
}
```

Q. 2 Write a C program to add two numbers using pointers.

Answer :-

```
#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 the numbers = %d\n", sum);
```

```
    return 0;
```

```
}
```

---

Q. 3 Write a C program to swap two numbers using pointers.

Answer :-

```
#include<stdio.h>

int main()
{
    int a= 10 ,b= 20;
    int *p=&a,*q=&b;
    int temp;
    printf("befor swapping a= %d and b= %d \n",*p,*q);

    temp = *p;
    *p=*q;
    *q=temp;
    printf("After swapping a= %d and b= %d \n",*p,*q);

}
```

Q. 4 Write a C program to input and print array elements using pointer.

Answer- 4

```
#include <stdio.h>

int main() {
    int n;
    printf("Enter the number of elements: ");
```

---

```
scanf("%d", &n);

int arr[n];
printf("Enter %d elements: \n", n);
for (int i = 0; i < n; i++) {
    scanf("%d", &arr[i]);
}

int *ptr = arr;
printf("The elements of the array are: ");
for (int i = 0; i < n; i++) {
    printf("%d ", *(ptr + i));
}

return 0;
}
```

Q. 5 Write a C program to copy one array to another using pointer.

Answer :-

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

```
void copy_array(int *source, int *destination, int size) {
    for (int i = 0; i < size; i++) {
        *destination = *source;
        source++;
    }
}
```

---

```
destination++;

}

}

int main() {
    int source_array[] = {1, 2, 3, 4, 5};
    int destination_array[5];

    copy_array(source_array, destination_array, 5);

    for (int i = 0; i < 5; i++) {
        printf("%d ", destination_array[i]);
    }

    return 0;
}
```

Q. 6 Write a C program to swap two arrays using pointers.

Answer :-

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

```
void swap_arrays(int *array1, int *array2, int size) {
    int temp;

    for (int i = 0; i < size; i++) {
        temp = array1[i];
```

---

```
array1[i] = array2[i];
array2[i] = temp;
}

}

int main() {
    int array1[] = {1, 2, 3, 4, 5};
    int array2[] = {6, 7, 8, 9, 10};
    int size = sizeof(array1) / sizeof(array1[0]);

    swap_arrays(array1, array2, size);

    printf("Array1: ");
    for (int i = 0; i < size; i++) {
        printf("%d ", array1[i]);
    }
    printf("\n");

    printf("Array2: ");
    for (int i = 0; i < size; i++) {
        printf("%d ", array2[i]);
    }
    printf("\n");
```

---

```
    return 0;
```

```
}
```

Q. 7 Write a C program to reverse an array using pointers.

Answer :-

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

```
int main() {
```

```
    int arr[] = {1, 2, 3, 4, 5};
```

```
    int *ptr1 = arr;
```

```
    int *ptr2 = arr + sizeof(arr) / sizeof(arr[0]) - 1;
```

```
    while (ptr1 < ptr2) {
```

```
        int temp = *ptr1;
```

```
        *ptr1 = *ptr2;
```

```
        *ptr2 = temp;
```

```
        ptr1++;
```

```
        ptr2--;
```

```
}
```

```
for (int i = 0; i < sizeof(arr) / sizeof(arr[0]); i++) {
```

```
    printf("%d ", arr[i]);
```

```
}
```

```
return 0;
```

---

```
}
```

Q. 8 Write a C program to add two matrix using pointers.

Answer:-

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

```
int main() {
```

```
    // Define the size of the matrices
```

```
    int m, n;
```

```
    printf("Enter the number of rows and columns of the matrices: ");
```

```
    scanf("%d%d", &m, &n);
```

```
    // Declare the matrices
```

```
    int a[m][n], b[m][n], c[m][n];
```

```
    // Get the elements of the matrices
```

```
    printf("Enter the elements of matrix A: \n");
```

```
    for (int i = 0; i < m; i++) {
```

```
        for (int j = 0; j < n; j++) {
```

```
            scanf("%d", &a[i][j]);
```

```
        }
```

```
}
```

```
    printf("Enter the elements of matrix B: \n");
```

```
    for (int i = 0; i < m; i++) {
```

---

```
for (int j = 0; j < n; j++) {  
    scanf("%d", &b[i][j]);  
}  
}  
  
// Add the matrices  
  
int *ptr1 = a[0];  
int *ptr2 = b[0];  
int *ptr3 = c[0];  
for (int i = 0; i < m * n; i++) {  
    *ptr3 = *ptr1 + *ptr2;  
    ptr1++;  
    ptr2++;  
    ptr3++;  
}  
  
// Print the resulting matrix  
  
printf("The resulting matrix is: \n");  
for (int i = 0; i < m; i++) {  
    for (int j = 0; j < n; j++) {  
        printf("%d ", c[i][j]);  
    }  
    printf("\n");  
}
```

---

---

```
    return 0;
```

```
}
```

Q. 9 Write a C program to multiply two matrix using pointers.

Answer :-

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

```
#include <stdlib.h>
```

```
// Function to multiply two matrices
```

```
void multiply_matrices(int **a, int **b, int **c, int n) {
```

```
    int i, j, k;
```

```
    for (i = 0; i < n; i++) {
```

```
        for (j = 0; j < n; j++) {
```

```
            c[i][j] = 0;
```

```
            for (k = 0; k < n; k++) {
```

```
                c[i][j] += a[i][k] * b[k][j];
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

```
// Function to print a matrix
```

```
void print_matrix(int **a, int n) {
```

```
    int i, j;
```

```
    for (i = 0; i < n; i++) {
```

```
        for (j = 0; j < n; j++) {
```

```
            printf("%d ", a[i][j]);
```

```
        }
```

```
    printf("\n");
}

}

int main() {
    // Create two matrices
    int n = 3;
    int *a = (int *)malloc(n * sizeof(int *));
    int *b = (int *)malloc(n * sizeof(int *));
    int *c = (int *)malloc(n * sizeof(int *));
    for (int i = 0; i < n; i++) {
        a[i] = (int *)malloc(n * sizeof(int));
        b[i] = (int *)malloc(n * sizeof(int));
        c[i] = (int *)malloc(n * sizeof(int));
    }

    // Input the elements of the matrices
    printf("Enter the elements of the first matrix:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            scanf("%d", &a[i][j]);
        }
    }

    printf("Enter the elements of the second matrix:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            scanf("%d", &b[i][j]);
        }
    }
```

```
}

}

// Multiply the matrices
multiply_matrices(a, b, c, n);

// Print the product matrix
printf("The product matrix is:\n");
print_matrix(c, n);

// Free the allocated memory
for (int i = 0; i < n; i++) {
    free(a[i]);
    free(b[i]);
    free(c[i]);
}
free(a);
free(b);
free(c);

return 0;
}
```

---

# C- Programming Language

## Week – 9

### Programming Questions

Q. 1 Write a C program to Search string.

Answer :-

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(int argc, char *argv[]) {
    // Check if the user has provided the correct number of arguments
    if (argc != 3) {
        printf("Usage: %s <file> <string>\n", argv[0]);
        return 1;
    }

    // Open the file for reading
    FILE *file = fopen(argv[1], "r");
    if (file == NULL) {
        printf("Error: Could not open file %s\n", argv[1]);
        return 1;
    }

    // Read the file line by line
    char line[1024];
    while (fgets(line, sizeof(line), file)) {
```

---

```
// Check if the line contains the search string
if (strstr(line, argv[2])) {
    // Print the line
    printf("%s", line);
}

// Close the file
fclose(file);

return 0;
}
```

Q. 2 Write a C program to Reverse words in string.

Answer :-

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main(int argc, char *argv[]) {
    if (argc != 3) {
        printf("Usage: %s <file> <string>\n", argv[0]);
        return 1;
    }

    FILE *file = fopen(argv[1], "r");
    if (file == NULL) {
        printf("Error: Could not open file %s\n", argv[1]);
        return 1;
    }
```

---

```
}

// Read the file line by line
char line[1024];
while (fgets(line, sizeof(line), file)) {
    // Check if the line contains the search string
    if (strstr(line, argv[2])) {
        // Print the line
        printf("%s", line);
    }
}

// Close the file
fclose(file);

return 0;
}
```

Q. 3 Write a C program to count vowels, consonants, etc.

Answer :-

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

int main() {
    char str[100];
    int i, vCount = 0, cCount = 0, dCount = 0, sCount = 0;

    printf("Enter a string: ");
    scanf("%s", str);
```

```
for (i = 0; i < strlen(str); i++) {  
    if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u') {  
        vCount++;  
    } else if (str[i] >= 'a' && str[i] <= 'z') {  
        cCount++;  
    } else if (str[i] >= '0' && str[i] <= '9') {  
        dCount++;  
    } else if (str[i] == ' ') {  
        sCount++;  
    }  
}  
  
printf("Vowels: %d\n", vCount);  
printf("Consonants: %d\n", cCount);  
printf("Digits: %d\n", dCount);  
printf("Spaces: %d\n", sCount);  
  
return 0;  
}
```

Q. 4 Create a program to separate characters in a given string?

Answer:-

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

```
int main()  
{  
    char str[100];  
    int i;
```

```
printf("Enter a string: ");
scanf("%s", str);

printf("The characters of the string are: ");
for (i = 0; str[i] != '\0'; i++)
{
    printf("%c ", str[i]);
}

printf("\n");

return 0;
}
```

Q. 5 Write a program to take two strings from user and concatenate them also add a space between them using strcat() function.

Sample input: JAI

GLA

Sample output: JAI GLA

Q. 6 Write a C program to take a string from user and make it toggle its case i.e. lower case to upper case and upper case to lower case.

Sample Input: HElLo wOrlD

Sample output: heLlO WoRLd

Answer :-

#include <stdio.h>

#include <string.h>

int main()

```
{  
    char str[100];  
    int i;  
  
    printf("Enter a string: ");  
    gets(str);  
  
    for (i = 0; str[i] != '\0'; i++)  
    {  
        if (str[i] >= 'a' && str[i] <= 'z')  
        {  
            str[i] = str[i] - 32;  
        }  
        else if (str[i] >= 'A' && str[i] <= 'Z')  
        {  
            str[i] = str[i] + 32;  
        }  
    }  
  
    printf("The converted string is: %s", str);  
  
    return 0;  
}
```

Q. 7 Write a C program to take two strings as input from user and check they are identical or not without using string functions.

Sample input: Jai Gla  
Jai Gla

Sample output: Identical

Q. 8 Write a C program to take a list of a student's names from user by asking number of students and sort them alphabetical order.

Sample Input:

Bhisham

Jayant

Abhishek

Dhruv

Sample Output:

Abhishek

Bhisham

Dhruv

Jayant

Answer:-

```
#include <stdio.h>
#include <string.h>
char ch[50];
void ascendingOrder() {
    int i, j;
    char tem;
    int SLength = strlen(ch);
    for (i = 0; i < SLength - 1; i++) {
        for (j = i + 1; j < SLength; j++) {
            if (ch[i] > ch[j]) {
                tem = ch[i];
                ch[i] = ch[j];
                ch[j] = tem;
            }
        }
    }
}
```

---

```
        }
    }
}

int main() {
    printf("\n Enter a string that you want to be arranged in alphabetical order
: ");
    fgets(ch, 50, stdin);
    ascendingOrder();
    puts(ch);
    return 0;
}
```

# C- Programming Language

## Week – 10

### Programming Questions

Q. 1 Write a C program to find length of string using pointers.

Answer:1

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

```
int main() {
    char str[100];
    char *ptr = str;

    printf("Enter a string: ");
    scanf("%s", str);

    while (*ptr != '\0') {
        ptr++;
    }
}
```

```
int length = ptr - str;  
  
printf("The length of the string is %d\n", length);  
  
return 0;  
}
```

Q. 2 Write a C program to copy one string to another using pointer.

Answer : 2

```
#include <stdio.h>  
  
int main() {  
  
    char str1[] = "Hello, world!";  
    char *str2 = str1;  
  
    printf("The original string is: %s\n", str1);
```

---

```
while (*str2 != '\0') {
    *str2++;
}

*str2++ = '\0';

printf("The copied string is: %s\n", str2);

return 0;
}
```

Q. 3 Write a C program to concatenate two strings using pointers.

Answer :-

```
#include <stdio.h>

int main() {
    char str1[] = "Hello";
    char str2[] = " world!";
}
```

```
char *ptr1 = str1;  
char *ptr2 = str2;  
  
while (*ptr2 != '\0') {  
    *ptr1 = *ptr2;  
    ptr1++;  
    ptr2++;  
}  
  
*ptr1 = '\0';  
  
printf("%s\n", str1);  
  
return 0;  
}
```

Q. 4 Write a C program to compare two strings using pointers.

Q. 5 WAP to find largest among three numbers using pointe

---

## Answer

```
#include <stdio.h>

int main() {
    int num1, num2, num3;
    int *largest = &num1; // Initialize the pointer to the first number

    // Get the three numbers from the user
    printf("Enter three numbers: ");
    scanf("%d %d %d", &num1, &num2, &num3);

    // Compare the three numbers using the pointer
    if (*largest < num2) {
        largest = &num2;
    }
    if (*largest < num3) {
        largest = &num3;
    }

    // Print the largest number
    printf("The largest number is: %d\n", *largest);

    return 0;
}
```

Q. 6 WAP to find largest among three numbers using pointer.

Answer :-

```
#include <stdio.h>

int main() {
    int num1, num2, num3;
    int *ptr1 = &num1;
```

```
int *ptr2 = &num2;
int *ptr3 = &num3;

printf("Enter three numbers: ");
scanf("%d %d %d", ptr1, ptr2, ptr3);

if (*ptr1 > *ptr2 && *ptr1 > *ptr3) {
    printf("The largest number is %d\n", *ptr1);
} else if (*ptr2 > *ptr1 && *ptr2 > *ptr3) {
    printf("The largest number is %d\n", *ptr2);
} else {
    printf("The largest number is %d\n", *ptr3);
}

return 0;
}
```

Q. 7 WAP to find factorial of a number using pointer.

Answer :-

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

```
void factorial(int n, int *fact) {
    *fact = 1;
```

```
for (int i = 1; i <= n; i++) {  
    *fact *= i;  
}  
}  
  
int main() {  
    int num, fact;  
    printf("Enter a number: ");  
    scanf("%d", &num);  
    factorial(num, &fact);  
    printf("Factorial of %d is: %d\n", num, fact);  
    return 0;  
}
```

Q. 8 Write a program to print largest even number present in an array using pointer to an array.

Answer :-

Q. 9 WAP to find sum of elements of an array using array of pointer.

Amswer :-

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

```
int main() {
```

---

```
int arr[] = {10, 5, 8, 21, 17, 32, 12};  
int *ptr = arr;  
int largest_even = arr[0];  
  
for (int i = 0; i < sizeof(arr) / sizeof(arr[0]); i++) {  
    if (*ptr % 2 == 0 && *ptr > largest_even) {  
        largest_even = *ptr;  
    }  
    ptr++;  
}  
  
printf("The largest even number in the array is %d\n", largest_even);
```

return 0;

}

Q. 10 WAP to compute simple interest using pointers.

Answer: -

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

```
int main() {  
    // Declare variables  
    float principal, rate, time, simple_interest;
```

---

```
float *p_principal, *p_rate, *p_time, *p_simple_interest;

// Initialize pointers
p_principal = &principal;
p_rate = &rate;
p_time = &time;
p_simple_interest = &simple_interest;

// Get user input
printf("Enter principal amount: ");
scanf("%f", p_principal);

printf("Enter rate of interest: ");
scanf("%f", p_rate);

printf("Enter time: ");
scanf("%f", p_time);

// Calculate simple interest
*p_simple_interest = (*p_principal * *p_rate * *p_time) / 100;

// Display simple interest
```

---

```
printf("Simple interest: %f\n", *p_simple_interest);
```

```
return 0;
```

```
}
```

Q. 11 Write a program to print largest even number present in an array using pointer to an array.

Answer :-

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

```
int main() {
```

```
    int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

```
    int *ptr = arr; // pointer to the array
```

```
    int largest_even = arr[0]; // largest even number found so far
```

```
    // iterate over the array
```

```
    for (int i = 0; i < sizeof(arr) / sizeof(arr[0]); i++) {
```

```
        // check if the current element is even and larger than the largest even number  
        // found so far
```

```
        if (arr[i] % 2 == 0 && arr[i] > largest_even) {
```

```
            largest_even = arr[i]; // update the largest even number
```

```
        }
```

```
}
```

---

```
// print the largest even number
printf("The largest even number in the array is %d\n", largest_even);

return 0;
}
```

## C- Programming Language

### Week – 11

### Programming Questions

Q. 1 Write a C function to return the maximum of three integers.

Answer :-

```
#include<stdio.h>
int main()
{
    int max(int a, int b, int c) {
        if (a > b) {
            if (a > c) {
```

---

```
    return a;
} else {
    return c;
}
} else {
if (b > c) {
    return b;
} else {
    return c;
}
}
```

Q. 2 Write a C function to check if a given number is prime or not.

Answer :-

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

```
int is_prime(int n) {
for (int i = 2; i * i <= n; i++) {
if (n % i == 0) {
    return 0;
}
}
```

---

```
    return 1;  
}  
  
int main() {  
    int n;  
    printf("Enter a number: ");  
    scanf("%d", &n);  
  
    if (is_prime(n)) {  
        printf("%d is a prime number.\n", n);  
    } else {  
        printf("%d is not a prime number.\n", n);  
    }  
  
    return 0;  
}
```

Q. 3 Write a C function to compute the factorial of a non-negative integer.

Answer :-

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

```
unsigned long long factorial(unsigned int n) {
```

```
if (n == 0) {  
    return 1;  
} else {  
    return n * factorial(n - 1);  
}  
  
int main() {  
    unsigned int n;  
    printf("Enter a non-negative integer: ");  
    scanf("%u", &n);  
    printf("The factorial of %u is %llu\n", n, factorial(n));  
    return 0;  
}
```

Q. 4 Write a C function to swap the values of two integers in actual arguments.

ANSWER :-

```
#include <stdio.h>  
  
int main()  
{  
    int var1, var2, temp;  
    printf("Enter two integers \n");  
    scanf("%d%d", &var1, &var2);
```

---

```
    printf("Before Swappingn First variable = %d\nSecond variable = %d \n", var1,
var2);
    temp = var1;
    var1 = var2;
    var2 = temp;
    printf("After Swappingn First variable = %d\nSecond variable = %d\n", var1,
var2);
    return 0;
}
```

Q. 5 Write a C function to compute the sum and average of an array of integers.

Answer :-

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

```
int main(){

    int arr[100], size, sum;
    float avg;

    printf("Enter the size of the array: ");
    scanf("%d", &size);

    printf("Enter the array elements: ");
```

```
for(int i = 0; i < size; i++){
    scanf("%d", &arr[i]);
}

// Set the initial sum to zero
sum = 0;

// Loop through the array to calculate the sum
for(int i = 0; i < size; i++){
    sum = sum + arr[i];
}

// Calculate the avg
avg = sum / size;

printf("Sum of array elements is: %d", sum);
printf("\nAvg. of arrays elements is: %.2f", avg);

return 0;

}
```

**Q. 6** Write a C function to find the GCD (Greatest Common Divisor) of two nonnegative integers using Euclid's algorithm.

Answer :-

**Q. 7** Write a C function to check if a given string is a valid palindrome, considering only alphanumeric characters and ignoring cases.

Answer :-

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

```
#include <string.h>
```

```
int is_palindrome(char str[]) {  
    int i, j;  
    int len = strlen(str);  
  
    // Convert the string to lowercase  
    for (i = 0; i < len; i++) {  
        str[i] = tolower(str[i]);  
    }
```

```
// Remove all non-alphanumeric characters  
for (i = 0, j = 0; i < len; i++) {  
    if (isalnum(str[i])) {
```

```
str[j++] = str[i];  
}  
}  
  
len = j;  
  
// Check if the string is a palindrome  
for (i = 0, j = len - 1; i < j; i++, j--) {  
    if (str[i] != str[j]) {  
        return 0;  
    }  
}  
  
return 1;  
}  
  
int main() {  
    char str[] = "Madam";  
  
    if (is_palindrome(str)) {  
        printf("The string is a palindrome.\n");  
    }  
}
```

```
    } else {
        printf("The string is not a palindrome.\n");
    }

    return 0;
}
```

Q. 8 Write a C function to calculate the sum and difference of two complex numbers.

#### H.O.T.S Questions

Q. 9 Write a C function to find the second largest and second smallest elements in an array of integers.

Answer :-

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

```
void find_second_largest_and_smallest(int arr[], int n, int *second_largest, int
                                         *second_smallest) {
    *second_largest = arr[0];
    *second_smallest = arr[0];

    for (int i = 1; i < n; i++) {
        if (arr[i] > *second_largest) {
```

```
*second_smallest = *second_largest;
*second_largest = arr[i];
} else if (arr[i] < *second_smallest) {
    *second_smallest = arr[i];
}
}
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int n = sizeof(arr) / sizeof(arr[0]);

    int second_largest, second_smallest;
    find_second_largest_and_smallest(arr, n, &second_largest, &second_smallest);

    printf("The second largest element is %d\n", second_largest);
    printf("The second smallest element is %d\n", second_smallest);

    return 0;
}
```

Q. 10 Write a C function to find the number of occurrences of each unique element in an array.

---

Answer :

```
#include <stdio.h>
int main()
{
    int num,i,j;
    printf("Enter the number of elements in the array : ");
    scanf("%d",&num);
    int arr[num];
    printf("Enter the elements of the array : ");
    for(i=0;i<num;i++)
    {
        scanf("%d",&arr[i]);
    }
    int frequency[num];
    int visited = -1;

    for(i = 0; i < num; i++)
    {
        int count = 1;
        for(j = i+1; j < num; j++)
        {
```

```
if(arr[i] == arr[j])
{
    count++;
    frequency[j] = visited;
}
if(frequency[i] != visited)
{
    frequency[i] = count;
}
printf("Element Frequency\n");
for(i = 0; i < num; i++)
{
    if(frequency[i] != visited)
    {
        printf("%d \t : %d \n", arr[i], frequency[i]);
    }
}
return 0;
}
```