INDEX

Programing in C

1. **Program to take the input from user to read integer, float, char values and print them.**

```c
#include<stdio.h>

int main(){

   int a;
   float b;
   char c;

  printf("Give Input for Integer value \n");
  scanf("%d", &a);
  printf("Output : %d \n", a);

  printf("Give Input for Float value \n");
  scanf("%f", &b);
  printf("Output : %f \n", b);

  printf("Give Input for Character value \n");
  scanf(" %c", &c);
  printf("Output : %c \n", c);

 return 0;

}
```

Output :

```
Give Input for Integer value
5
Output : 5
Give Input for Float value
5.5
Output : 5.500000
Give Input for Character value
V
Output : V
```

2. Program to write ASCII value of character.

```c
#include <stdio.h>

int main() {
    char character;

    printf("Enter a character: ");
    scanf("%c", &character);

    printf("The ASCII value of '%c' is %d\n", character, character);

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./ASCII
Enter a character: v
The ASCII value of 'v' is 118
```

3. Program to use getchar() and putchar(), gets and puts functions.

```c
#include <stdio.h>

int main() {
    char ch;
    char str[100];

    // Using getchar() and putchar()
    printf("Enter a single character: ");
    ch = getchar(); // Reads a single character
    printf("You entered: ");
    putchar(ch); // Outputs the single character
    putchar('\n'); // Print a newline for formatting

    // Clearing input buffer
    while (getchar() != '\n'); // Discard extra input if any

    // Using gets() and puts()
    printf("Enter a string (up to 99 characters): ");
    gets(str); // Reads a string
    printf("You entered: ");
    puts(str); // Outputs the string

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./GetcharPutchar
Enter a single character: V
You entered: V
Enter a string (up to 99 characters): Vishal
You entered: Vishal
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

4. Program to Check Vowels in a String.

```c
#include <stdio.h>

int main() {
    char str[100];
    int hasVowel = 0;

    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);

    for (int i = 0; str[i] != '\0'; i++) {
        char ch = str[i];
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||
            ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {
            hasVowel = 1;
            break;
        }
    }

    if (hasVowel) {
        printf("The string contains vowels.\n");
    } else {
        printf("The string does not contain any vowels.\n");
    }

    return 0;
}
```

Output :

```
Enter a string (up to 99 characters): Vishal
You entered: Vishal
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Vowels
Enter a string: Vishal
The string contains vowels.
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

5. Program to reverse of characters.

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

int main() {
    char str[100], temp;
    char *left, *right;

    printf("Enter a string: ");
    scanf("%99s", str);

    left = str;
    right = str + strlen(str) - 1;

    while (left < right) {
        temp = *left;
        *left = *right;
        *right = temp;

        left++;
        right--;
    }

    printf("Reversed string: %s\n", str);

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./CharacterReverse
Enter a string: Vishal
Reversed string: lahsiV
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

6. Largest and Smallest Number in an Array.

```c
#include<stdio.h>

void Largest(int arr[], int len){
   int left = 0;
   int right = len-1;

   while (left<=right)
   {
    if (arr[left]>arr[right])
    {
       right--;
    }else{
       left++;
    }
   }
   printf("Largest %d \n", arr[right]);
}

void Smallest(int arr[], int len){
   int left = 0;
   int right = len-1;

   while (left<=right)
   {
    if (arr[left]<arr[right])
    {
       right--;
    }else{
       left++;
    }
   }
   printf("Smallest %d \n", arr[right]);
}

int main(){
int arr[] = {100,56,756,42,77,2,66,90};
int len = sizeof(arr)/sizeof(arr[0]);
Largest(arr, len);
Smallest(arr, len);
return 0;
}
```

Output :

```
Largest 756
Smallest 2

[Done] exited with code=0 in 0.959 seconds
```

7. Largest and Smallest Using Global Declaration.

```c
#include<stdio.h>

int main(){
int a;
int b;

printf("Enter first number : \n");
scanf("%d", &a);

printf("Enter Second number : \n");
scanf("%d", &b);

if (a>b)
{
    printf("%d is Larger than %d", a , b);
}else{
    printf("%d is Larger than %d", b, a);
}

}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Largest
Enter first number :
5
Enter Second number :
7
7 is Larger than 5
```

8. Factorial of a number.

```c
#include<stdio.h>

int main(){

    int num;
    printf("Enter the number you want factorial of : ");
    scanf("%d", &num);
    fflush(stdin);

    int fact = 1;

    while (num!=0)
    {
        fact = fact*num--;
    }

    printf("%d \n", fact);

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Factorial
Enter the number you want factorial of : 5
120
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

9. Fibonacci Series.

```c
#include <stdio.h>

int main() {
    int len;

    printf("Enter the Length of Fibonacci: ");
    scanf("%d", &len);

    int first = 0;
    int second = 1;
    int next;

    if (len == 1) {
        printf("%d\n", first);
    }

    else if (len == 2) {
        printf("%d, %d\n", first, second);

    }

    else if (len > 2) {
        printf("%d, %d", first, second);

        for (int i = 2; i < len; i++) {
            next = first + second;
            first = second;
            second = next;
            printf(", %d", next);
        }
        printf("\n");
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Fibbonacci
Enter the Length of Fibonacci: 5
0, 1, 1, 2, 3
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

10. Palindrome of a number.

```c
#include<stdio.h>

void main(){

    int num;
    printf("Enter the number you want to check Palindrome of : ");
    scanf("%d", &num);

    int palindrome = 0;
    int temp = num;

    while (num != 0)
    {
       int digit = num%10;
       palindrome = palindrome*10 + digit;
       num = num/10;
    }

    if (temp == palindrome)
    {
       printf("Yes the number is a Palindrome! ");
    }else{
       printf("Number is not a Palindrome! ");
    }

}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Palindrome
Enter the number you want to check Palindrome of : 505
Yes the number is a Palindrome!
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

11. Sum of 10 values using array.

```c
#include<stdio.h>

void main(){
    int arr[] = {1,2,3,4,5,6,7,8,9,10};
    int sum =0;
    for (int i = 0; i < 10; i++)
    {
        sum += arr[i];
    }
    printf("%d \n", sum);
}
```

Output :

```
[Running] cd "c:\Users\hp\Desktop\C_Programs\Arrays\" && gcc SumOfArray.c -o SumOfArray &&
"c:\Users\hp\Desktop\C_Programs\Arrays\"SumOfArray
55
```

12. Average of n values.

```c
#include <stdio.h>

void main()
{
    int n;
    printf("Enter the Length of Array \n");
    scanf("%d", &n);
    int arr[n];

    int avg = 0;

    for (int i = 1; i <= n; i++)
    {
        printf("Enter the %d Element: ", i);
        scanf("%d", &arr[i]);
    }

    int sum =0;
    for (int i = 1; i <= n; i++)
    {
        sum += arr[i];
    }
    avg = sum/n;

    printf("Avergae :  %d \n", avg);
```

```
}
```

Output :

```
Enter the Length of Array
5
Enter the 1 Element: 11
Enter the 2 Element: 23
Enter the 3 Element: 645
Enter the 4 Element: 34
Enter the 5 Element: 13
Avergae :  145
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

13. Armstrong Number.

```c
#include<stdio.h>

int main(){
    int num;
    printf("Enter the number : ");
    scanf("%d", &num);

    int armstrong = 0;
    int temp = num;

    while (num!=0)
    {
        int digit = num%10;
        armstrong += digit*digit*digit;
        num = num/10;
    }

    if (armstrong == temp)
    {
        printf("Number is an Armstrong");
    }else{
        printf("Number is not an Armstrong");
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Armstrong
Enter the number : 555
Number is not an Armstrong
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Armstrong
Enter the number : 153
Number is an Armstrong
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

14. Odd And Even.

```c
#include<stdio.h>

void main(){

    int num;
    printf("Enter the num ");
    scanf("%d", &num);

    if (num%2==0)
    {
       printf("Number is even");
    }else{
       printf("Number is Odd");
    }

}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./OddAndEven
Enter the num 55
Number is Odd
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./OddAndEven
Enter the num 44
Number is even
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

15. Factors of a Number.

```c
#include <stdio.h>

void findFactors(int num) {
    printf("Factors of %d are: ", num);
    for (int i = 1; i <= num; i++) {
        if (num % i == 0) {
            printf("%d ", i);
        }
    }
    printf("\n");
}

int main() {
    int number;

    printf("Enter a number: ");
    scanf("%d", &number);

    if (number <= 0) {
        printf("Please enter a positive number.\n");
    } else {
        findFactors(number);
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Loops> ./Factors
Enter a number: 20
Factors of 20 are: 1 2 4 5 10 20
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

16. First n Prime Numbers.

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

int isPrime(int num) {
    if (num <= 1) return 0;
    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) return 0;
    }
    return 1;
}

void printFirstNPrimes(int n) {
    int count = 0, num = 2;
    printf("The first %d prime numbers are: ", n);
    while (count < n) {
        if (isPrime(num)) {
            printf("%d ", num);
            count++;
        }
        num++;
    }
    printf("\n");
}

int main() {
    int n;

    printf("Enter the number of prime numbers to display: ");
    scanf("%d", &n);

    if (n <= 0) {
        printf("Please enter a positive integer.\n");
    } else {
        printFirstNPrimes(n);
    }

    return 0;
}
```

Output :

```
Enter the number of prime numbers to display: 5
The first 5 prime numbers are: 2 3 5 7 11
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

17. Whether the Enter the Number is float or int.

```c
#include <stdio.h>

int main() {
    float num;

    printf("Enter a number: ");

    if (scanf("%f", &num) == 1) {
        // If the number has a decimal point, it's a float
        if ((int)num != num) {
            printf("The number is a float.\n");
        } else {
            printf("The number is an integer.\n");
        }
    } else {
        printf("Invalid input!\n");
    }

    return 0;
}
```

Output :

```
Enter a number: 5
The number is an integer.
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./FloatorInt
Enter a number: 5.5
The number is a float.
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

18. Table of a Number.

```c
#include<stdio.h>

int main(){
    int num;
    printf("Enter the number: ");
    scanf("%d", &num);

    for (int i = 1; i <= 10; i++)
    {
        printf("%d * %d = %d \n",num,i,num*i);
    }

}
```

Output :

```
Enter the number: 2
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
2 * 10 = 20
PS C:\Users\hp\Desktop\C_Programs\Loops>
```

19. Reverse of an Array.

```c
#include<stdio.h>

void ReverseOfArray(int arr[],int n){
    int left = 0;
    int right = n-1;

    while (left<right)
    {
        int temp = arr[left];
        arr[left] = arr[right];
        arr[right] = temp;

        left++;
        right--;
    }

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

int main(){
    int n;
    printf("Enter the Length of Array: ");
    scanf("%d", &n);
    int arr[n];

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

    ReverseOfArray(arr, n);

    return 0;
}
```

Output :

```
Enter the Length of Array: 5
Enter the Elements of array:
1
2
3
4
5
Array Reversed : 5 4 3 2 1
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

20. Insert an Element into Array.

```c
#include <stdio.h>

int main() {
    int n, position, element;

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

    int arr[n + 1];

    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }

    printf("Enter the element to insert: ");
    scanf("%d", &element);
    printf("Enter the position to insert (1 to %d): ", n + 1);
    scanf("%d", &position);

    if (position < 1 || position > n + 1) {
        printf("Invalid position!\n");
        return 1;
    }

    for (int i = n; i >= position; i--) {
        arr[i] = arr[i - 1];
    }


    arr[position - 1] = element;

    printf("The updated array is:\n");
    for (int i = 0; i <= n; i++) {
```

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

    printf("\n");

    return 0;
}
```

Output :

```
Enter the size of the array: 5
Enter 5 elements:
Element 1: 1
Element 2: 5
Element 3: 3
Element 4: 6
Element 5: 4
Enter the element to insert: 44
Enter the position to insert (1 to 6): 4
The updated array is:
1 5 3 44 6 4
PS C:\Users\hp\Desktop\C_Programs\DataStructure>
```

21. Delete an Element from an Array.

```
#include <stdio.h>

int main() {
    int n, position;

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

    int arr[n];

    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }

    printf("Enter the position of the element to delete (1 to %d): ", n);
    scanf("%d", &position);

    if (position < 1 || position > n) {
        printf("Invalid position!\n");
        return 1;
```

```
    }


    for (int i = position - 1; i < n - 1; i++) {
        arr[i] = arr[i + 1];
    }

    printf("The updated array is:\n");
    for (int i = 0; i < n - 1; i++) {
        printf("%d ", arr[i]);
    }

    printf("\n");

    return 0;
}
```

Output :

```
Enter the size of the array: 5
Enter 5 elements:
Element 1: 1
Element 2: 2
Element 3: 4
Element 4: 5
Element 5: 7
Enter the position of the element to delete (1 to 5): 4
The updated array is:
1 2 4 7
```

22. Linear Search.

```c
#include<stdio.h>

int LinearSearch(int arr[], int len, int target){
    for (int i = 0; i < len; i++)
    {
        if (arr[i] == target)
        {
            printf("Target found and available at index : ");
            return i;
        }
    }
    return -1;
}

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

    int arr[n];

    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }
    int target;
    printf("Enter the target : ");
    scanf("%d", &target);

    int result = LinearSearch(arr,n,target);

    printf("%d", result);

    return 0;
    }
```

Output :

```
Enter the size of the array: 5
Enter 5 elements:
Element 1: 11
Element 2: 23
Element 3: 12
Element 4: 55
Element 5: 6
Enter the target : 12
Target found and available at index : 2
PS C:\Users\hp\Desktop\C_Programs\DataStructure>
```

23. Binary Search.

```c
#include<stdio.h>

int BinarySearch(int arr[], int len, int target){
    int left =0;
    int right =len-1;

    while (left <= right)
    {
        int mid = left + (right - left)/2;

        if (target == arr[mid])
        {
            printf("Target found and available at index : ");
            return mid;
        }

        else if (target < arr[mid])
        {
            right = mid - 1;
        }
        else{
            left = mid + 1;
        }
    }
      return -1;

}

int main(){
    int n;

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

```
    int arr[n];

    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }

    int target;
    printf("Enter the target : ");
    scanf("%d", &target);

    int result = BinarySearch(arr,n,target);
    printf("%d", result);

    return 0;
}
```

Output :

```
Enter the size of the array: 5
Enter 5 elements:
Element 1: 12
Element 2: 32
Element 3: 65
Element 4: 34
Element 5: 76
Enter the target : 65
Target found and available at index : 2
PS C:\Users\hp\Desktop\C_Programs\DataStructure>
```

24. Sorting List using Array.

```
#include<stdbool.h>

int main(){

int n;
printf("Enter the Length of array \n");
scanf("%d", &n);

int arr[n];

for (int k = 0; k < n; k++)
{
    printf("%d Element of array : ", k+1);
    scanf("%d", &arr[k]);
}
```

```
bool swap = true;

while (swap)
{
    swap = false;
    for (int i = 0; i<n-1; i++)
    {
        if (arr[i] > arr[i+1])
        {
           int temp = arr[i];
           arr[i] = arr[i+1];
           arr[i+1] = temp;
            swap = true;
        }
    }
    if (swap == false)
    {
        break;
    }
}

printf("Array after Bubble sort \n");
for (int k = 0; k < n; k++)
{
    printf("%d ", arr[k]);
}

}
```

Output :

```
Enter the Length of array
5
1 Element of array : 12
2 Element of array : 42
3 Element of array : 2
4 Element of array : 11
5 Element of array : 4
Array after Bubble sort
2 4 11 12 42
PS C:\Users\hp\Desktop\C_Programs\DataStructure>
```

25. Read and Print Square Matrix.

```c
#include <stdio.h>

int main() {
    int n;

    // Input size of the square matrix
    printf("Enter the size of the square matrix (n): ");
    scanf("%d", &n);

    int matrix[n][n];

    // Reading the matrix elements
    printf("Enter the elements of the %dx%d matrix:\n", n, n);
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("Element [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &matrix[i][j]);
        }
    }

    // Printing the matrix
    printf("The entered matrix is:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("%d ", matrix[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

Output :

```
Enter the size of the square matrix (n): 3
Enter the elements of the 3x3 matrix:
Element [1][1]: 1
Element [1][2]: 2
Element [1][3]: 3
Element [2][1]: 4
Element [2][2]: 5
Element [2][3]: 6
Element [3][1]: 7
Element [3][2]: 8
Element [3][3]: 9
The entered matrix is:
1 2 3
4 5 6
7 8 9
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

26. Addison and Subtraction in 2 matrixes.

```c
#include <stdio.h>

int main() {
    int n, m;

    printf("Enter the number of rows and columns of the matrices: ");
    scanf("%d %d", &n, &m);

    int matrix1[n][m], matrix2[n][m], sum[n][m], difference[n][m];

    printf("Enter elements of the first matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            scanf("%d", &matrix1[i][j]);

    printf("Enter elements of the second matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            scanf("%d", &matrix2[i][j]);

    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++) {
            sum[i][j] = matrix1[i][j] + matrix2[i][j];
            difference[i][j] = matrix1[i][j] - matrix2[i][j];
        }

    printf("Sum of the matrices:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++)
            printf("%d ", sum[i][j]);
        printf("\n");
    }
```

```
    printf("Difference of the matrices:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++)
            printf("%d ", difference[i][j]);
        printf("\n");
    }

    return 0;
}
```

Output :

```
Enter the number of rows and columns of the matrices: 2
2
Enter elements of the first matrix:
1
2
3
4
Enter elements of the second matrix:
5
6
7
8
Sum of the matrices:
6 8
10 12
Difference of the matrices:
-4 -4
-4 -4
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

27. Transpose of Matrix.

```
#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], transpose[cols][rows];

    printf("Enter elements of the matrix:\n");
    for (int i = 0; i < rows; i++)
```

```
        for (int j = 0; j < cols; j++)
            scanf("%d", &matrix[i][j]);

    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            transpose[j][i] = matrix[i][j];

    printf("Original matrix:\n");
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++)
            printf("%d ", matrix[i][j]);
        printf("\n");
    }

    printf("Transpose of the matrix:\n");
    for (int i = 0; i < cols; i++) {
        for (int j = 0; j < rows; j++)
            printf("%d ", transpose[i][j]);
        printf("\n");
    }

    return 0;
}
```

Output :

```
Enter the number of rows and columns of the matrix: 3
3
Enter elements of the matrix:
1
2
3
4
5
6
7
8
9
Original matrix:
1 2 3
4 5 6
7 8 9
Transpose of the matrix:
1 4 7
2 5 8
3 6 9
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

28. Multiply two matrixes.

```c
#include <stdio.h>

int main() {
    int r1, c1, r2, c2;

    printf("Enter the number of rows and columns of the first matrix: ");
    scanf("%d %d", &r1, &c1);

    printf("Enter the number of rows and columns of the second matrix: ");
    scanf("%d %d", &r2, &c2);

    if (c1 != r2) {
        printf("Matrix multiplication is not possible. Number of columns in
the first matrix must equal the number of rows in the second matrix.\n");
        return 0;
    }

    int matrix1[r1][c1], matrix2[r2][c2], result[r1][c2];

    printf("Enter elements of the first matrix:\n");
    for (int i = 0; i < r1; i++)
        for (int j = 0; j < c1; j++)
            scanf("%d", &matrix1[i][j]);

    printf("Enter elements of the second matrix:\n");
    for (int i = 0; i < r2; i++)
        for (int j = 0; j < c2; j++)
            scanf("%d", &matrix2[i][j]);

    for (int i = 0; i < r1; i++)
        for (int j = 0; j < c2; j++) {
            result[i][j] = 0;
            for (int k = 0; k < c1; k++)
                result[i][j] += matrix1[i][k] * matrix2[k][j];
        }

    printf("Resultant matrix after multiplication:\n");
    for (int i = 0; i < r1; i++) {
        for (int j = 0; j < c2; j++)
            printf("%d ", result[i][j]);
        printf("\n");
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\Arrays> ./MultiplyMatix
Enter the number of rows and columns of the first matrix: 2 2
Enter the number of rows and columns of the second matrix: 2 2
Enter elements of the first matrix:
1
2
3
4
Enter elements of the second matrix:
1
2
3
4
Resultant matrix after multiplication:
7 10
15 22
PS C:\Users\hp\Desktop\C_Programs\Arrays>
```

29. Leap Year.

```c
#include<stdio.h>

int main(){
    int year;
    printf("Enter the Year : ");
    scanf("%d", &year);

    if (year % 4 == 0)
    {
        printf("Year %d is a leap Year", year);
    }else{
        printf("Year %d is not a leap Year", year);
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./leapYear
Enter the Year : 2025
Year 2025 is not a leap Year
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

30. Largest of three numbers.

```c
#include<stdio.h>

int main(){
int a;
int b;
int c;

printf("Enter first number : \n");
scanf("%d", &a);

printf("Enter Second number : \n");
scanf("%d", &b);

printf("Enter Third number : \n");
scanf("%d", &c);

if (a>b && a>c)
{
    printf("%d is Larger than %d and %d", a , b, c);
}else if(b>a && b>c){
    printf("%d is Larger than %d and %d", b, a, c);
}else if(c>a && c>b){
    printf("%d is Larger than %d and %d", c, a, b);
}else {
    printf("All the numbers are same");
}

}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Largest
Enter first number :
22
Enter Second number :
33
Enter Third number :
445
445 is Larger than 22 and 33
```

31. Second Largest among Three.

```c
#include<stdio.h>

int main(){
int a;
int b;
int c;

printf("Enter first number : \n");
scanf("%d", &a);

printf("Enter Second number : \n");
scanf("%d", &b);

printf("Enter Third number : \n");
scanf("%d", &c);

if (a>b && a<c)
{
    printf("%d is Second largest", a);
}else if(b>a && b<c){
     printf("%d is Second largest", b);
}else if(c>a && c<b){
     printf("%d is Second largest", c);
}else {
    printf("All the numbers are same");
}

}
```

Output :

```
Enter first number :
22
Enter Second number :
11
Enter Third number :
66
22 is Second largest
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

32. Area of Different Shapes.

```c
#include<stdio.h>

void AreaOfCircle(){
int radius = 4;
float pi = 3.14;
float area = pi*radius*radius;
printf("Area of circle : %f \n", area);
}

void AreaOfRectangle(int length, int width){
int area = length * width;
printf("Area of Rectangle : %d \n", area);
}

int main(){
int length;
printf("Enter the Length for rectangle ");
scanf("%d", &length);
int width;
printf("Enter the Width for rectangle ");
scanf("%d", &width);

AreaOfCircle();
AreaOfRectangle(length, width);
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./AreaOfDifferentShapes
Enter the Length for rectangle 5
Enter the Width for rectangle 6
Area of circle : 50.240002
Area of Rectangle : 30
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

33. Calculator.

```c
#include <stdio.h>

int main() {
    char operator;
    double num1, num2, result;

    printf("Enter an operator (+, -, *, /): ");
    scanf(" %c", &operator);

    printf("Enter two numbers: ");
    scanf("%lf %lf", &num1, &num2);

    switch (operator) {
        case '+':
            result = num1 + num2;
            printf("Result: %.2lf\n", result);
            break;
        case '-':
            result = num1 - num2;
            printf("Result: %.2lf\n", result);
            break;
        case '*':
            result = num1 * num2;
            printf("Result: %.2lf\n", result);
            break;
        case '/':
            if (num2 != 0) {
                result = num1 / num2;
                printf("Result: %.2lf\n", result);
            } else {
                printf("Error: Division by zero is not allowed.\n");
            }
            break;
        default:
            printf("Error: Invalid operator.\n");
    }

    return 0;
}
```

Output :

```
Enter an operator (+, -, *, /): *
Enter two numbers: 5
5
Result: 25.00
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Calculator
Enter an operator (+, -, *, /): +
Enter two numbers: 33 44
Result: 77.00
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Calculator
Enter an operator (+, -, *, /): -
Enter two numbers:  40 9
Result: 31.00
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Calculator
Enter an operator (+, -, *, /): /
Enter two numbers: 22 10
Result: 2.20
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

34. Quadratic Equation.

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

int main() {
    double a, b, c, discriminant, root1, root2, realPart, imaginaryPart;

    printf("Enter coefficients a, b, and c: ");
    scanf("%lf %lf %lf", &a, &b, &c);

    if (a == 0) {
        printf("This is not a quadratic equation.\n");
        return 0;
    }

    discriminant = b * b - 4 * a * c;

    if (discriminant > 0) {
        root1 = (-b + sqrt(discriminant)) / (2 * a);
        root2 = (-b - sqrt(discriminant)) / (2 * a);
        printf("Roots are real and distinct:\n");
        printf("Root 1 = %.2lf\n", root1);
        printf("Root 2 = %.2lf\n", root2);
    } else if (discriminant == 0) {
        root1 = root2 = -b / (2 * a);
        printf("Roots are real and equal:\n");
        printf("Root = %.2lf\n", root1);
    } else {
```

```
        realPart = -b / (2 * a);
        imaginaryPart = sqrt(-discriminant) / (2 * a);
        printf("Roots are complex and imaginary:\n");
        printf("Root 1 = %.2lf + %.2lfi\n", realPart, imaginaryPart);
        printf("Root 2 = %.2lf - %.2lfi\n", realPart, imaginaryPart);
    }

    return 0;
}
```

Output :

```
Enter coefficients a, b, and c: 2 3 5
Roots are complex and imaginary:
Root 1 = -0.75 + 1.39i
Root 2 = -0.75 - 1.39i
```

35. Compound and Simple Interest.

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

int main() {
    double principal, rate, time, simpleInterest, compoundInterest;

    printf("Enter principal amount: ");
    scanf("%lf", &principal);

    printf("Enter annual interest rate (in percentage): ");
    scanf("%lf", &rate);

    printf("Enter time (in years): ");
    scanf("%lf", &time);

    simpleInterest = (principal * rate * time) / 100;

    compoundInterest = principal * pow((1 + rate / 100), time) - principal;

    printf("Simple Interest = %.2lf\n", simpleInterest);
    printf("Compound Interest = %.2lf\n", compoundInterest);

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C Programs\SimpleQuestions> ./SIANDCI
Enter principal amount: 100
Enter annual interest rate (in percentage): 2
Enter time (in years): 2
Simple Interest = 4.00
Compound Interest = 4.04
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

36. Convert Celsius to Fahrenheit.

```c
#include <stdio.h>

int main() {
    int choice;
    double temperature, converted;

    printf("Temperature Conversion Menu:\n");
    printf("1. Convert Celsius to Fahrenheit\n");
    printf("2. Convert Fahrenheit to Celsius\n");
    printf("Enter your choice (1 or 2): ");
    scanf("%d", &choice);

    if (choice == 1) {
        printf("Enter temperature in Celsius: ");
        scanf("%lf", &temperature);
        converted = (temperature * 9 / 5) + 32;
        printf("Temperature in Fahrenheit: %.2lf\n", converted);
    } else if (choice == 2) {
        printf("Enter temperature in Fahrenheit: ");
        scanf("%lf", &temperature);
        converted = (temperature - 32) * 5 / 9;
        printf("Temperature in Celsius: %.2lf\n", converted);
    } else {
        printf("Invalid choice! Please select 1 or 2.\n");
    }

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./CelsiustoF
Temperature Conversion Menu:
1. Convert Celsius to Fahrenheit
2. Convert Fahrenheit to Celsius
Enter your choice (1 or 2): 1
Enter temperature in Celsius: 45
Temperature in Fahrenheit: 113.00
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./CelsiustoF
Temperature Conversion Menu:
1. Convert Celsius to Fahrenheit
2. Convert Fahrenheit to Celsius
Enter your choice (1 or 2): 2
Enter temperature in Fahrenheit: 110
Temperature in Celsius: 43.33
```

37. Diamond Pattern.

```c
// C program to print
// diamond shape with
// 2n rows
#include<stdio.h>

// Prints diamond
// pattern with 2n rows
void printDiamond(int n)
{
    int space = n - 1;

    // run loop (parent loop)
    // till number of rows
    for (int i = 0; i < n; i++)
    {
        // loop for initially space,
        // before star printing
        for (int j = 0;j < space; j++)
            printf(" ");

        // Print i+1 stars
        for (int j = 0;j <= i; j++)
            printf("* ");

        printf("\n");
        space--;
    }

    // Repeat again in
    // reverse order
```

```
    space = 0;

    // run loop (parent loop)
    // till number of rows
    for (int i = n; i > 0; i--)
    {
        // loop for initially space,
        // before star printing
        for (int j = 0; j < space; j++)
            printf(" ");

        // Print i stars
        for (int j = 0;j < i;j++)
            printf("* ");

        printf("\n");
        space++;
    }
}

// Driver code
int main()
{
    printDiamond(5);
    return 0;
}
```

Output :

```
[Running] cd "c:\Users\hp\Desktop\C_Programs\Pattern\" && gcc DiamondPattern.c -o DiamondP
"c:\Users\hp\Desktop\C_Programs\Pattern\"DiamondPattern
    *
   * *
  * * *
 * * * *
* * * * *
* * * * *
 * * * *
  * * *
   * *
    *
```

38. Number Triangle.

```c
#include<stdio.h>

int main(){
    int n= 5;
    for (int i = 1; i <= n; i++)
    {
        int print = 1;
        for (int j = 1; j <= i; j++)
        {
            printf("%d", print);
            print++;
        }
        printf("\n");
    }

}
```

Output :

```
1
12
123
1234
12345

[Done] exited with code=0 in 0.558 seconds
```

39. Pyramid.

```c
#include<stdio.h>

int main(){
    int n = 5;

    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= n-i; j++)
        {
            printf(" ");
        }
```

```
    for (int k = 1; k <= 2*i-1; k++)
    {
         printf("*");
    }
        printf("\n");
    }

}
```

Output :

```
[Running] cd "c:\Users\hp\Desktop\C_Programs\Pattern\" && gcc Pryamid.c -o
    *
   ***
  *****
 *******
*********
```

40. Invert Triangle.

```
#include<stdio.h>

int main(){
int n = 5;

for (int i = 5; i >= 1; i--)
{
    for (int j = 1; j <= i; j++)
    {
        printf("*");
    }
    printf("\n");

}
}
```

Output :

```
[Running] cd "c:\Users\hp\Desktop\C_Programs\Pattern\" && gcc InvertTraingle
"c:\Users\hp\Desktop\C_Programs\Pattern\"InvertTraingle
*****
****
***
**
*

[Done] exited with code=0 in 0.409 seconds
```

41. Read and Write Character Array.

```c
#include <stdio.h>

int main() {
    char str[100];

    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);  // Reading input string

    printf("You entered: %s\n", str);  // Writing/printing the string

    return 0;
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./ReadAndWriteArray
Enter a string: vishal
You entered: vishal
```

42. Marksheet of a Student.

```
#include<conio.h>
#include<stdio.h>
void main()
{
    char name, grade;
    float cn, c, dsa, csa, dbms, se, rollno, total, percentage;
    printf("MARKSHEET\n");
    printf("Name of the student:");
    scanf("%s",&name);
    printf("Roll No.:");
    scanf("%f",& rollno);
    printf("marks of Computer Networking:");
    scanf("%f",&cn);
    printf("marks of C language:");
    scanf("%f",&c);
    printf("marks of DSA:");
    scanf("%f",&dsa);
    printf("marks of CSA:");
    scanf("%f",&csa);
    printf("marks of DBMS:");
    scanf("%f",&dbms);
    printf("marks of Software Engineering:");
    scanf("%f",&se);
    total=cn+c+dsa+csa+dbms+se;
    percentage=(total/600)*100;
    printf("Total Marks:%f\n",total);
    printf("Percentage:%f\n",percentage);
    if (percentage>=90) {grade = 'A';}
    else if(percentage>=80) {grade = 'B';}
    else if(percentage>=70) {grade = 'C';}
    else if(percentage>=60) {grade = 'D';}
    else if(percentage>=35) {grade ='E';}
    else {grade = 'F';}
    printf("Grade:%c\n",grade);
}
```

Output :

```
MARKSHEET
Name of the student:Vishal
Roll No.:19
marks of Computer Networking:45
marks of C language:67
marks of DSA:55
marks of CSA:56
marks of DBMS:34
marks of Software Engineering:77
Total Marks:334.000000
Percentage:55.666668
Grade:E
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions>
```

43. Salary of an Employee.

```c
#include<conio.h>
#include<stdio.h>
void main()
{
    char name;
    int id;
    float basic_salary, allowances, deduction, gross_salary, net_salary;
    printf("Enter the name of employee: ");
    scanf("%s",&name);
    printf("Enter the id of employee: ");
    scanf("%d",&id);
    printf("Basic salary of employee: ");
    scanf("%f",&basic_salary);
    printf("Enter the total allowances: ");
    scanf("%f",&allowances);
    printf("Enter the total deduction: ");
    scanf("%f",&deduction);
    gross_salary=basic_salary+allowances;
    net_salary=gross_salary-deduction;
    printf("Salary Slip\n");
    printf("------------\n");
    printf("Name: %s\n",name);
    printf("Id:%d\n",id);
    printf("Basic salary: %f\n",basic_salary);
    printf("Allowances: %f\n",allowances);
    printf("Deduction: %f\n",deduction);
    printf("Gross salary %f\n",gross_salary);
    printf("Net salary %f\n",net_salary);
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./Salary
Enter the name of employee: Vishal
Enter the id of employee: 2
Basic salary of employee: 40000
Enter the total allowances: 35000
Enter the total deduction: 5000
Salary Slip
-----------
```

44. Bill Calculation.

```
#include<conio.h>
#include<stdio.h>
void main()
{
    float price, discount_rate, tax_rate, total, discount_amount, tax_amount,
final_amount;
    int quantity;
    printf("Enter the price: ");
    scanf("%f",&price);
    printf("Enter the quantity: ");
    scanf("%d",&quantity);
    total=price*quantity;
    printf("Enter the discount rate(in percentage): ");
    scanf("%f",&discount_rate);
    printf("Enter the tax rate(in percentage): ");
    scanf("%f",&tax_rate);
    discount_amount=(total*discount_rate)/100;
    tax_amount=(total*tax_rate)/100;

    final_amount=total+tax_amount-discount_amount;

    printf("Bill Summary\n ");
    printf("----------------\n");
    printf("price: %f\n ",price);
    printf("quantity: %d\n ",quantity);
    printf("total: %f\n ",total);
    printf("discount amount: %f\n ",discount_amount);
    printf("tax amount: %f\n ",tax_amount);
    printf("final amount: %f\n ",final_amount);
}
```

Output :

```
PS C:\Users\hp\Desktop\C_Programs\SimpleQuestions> ./BillSlips
Enter the price: 5000
Enter the quantity: 5
Enter the discount rate(in percentage): 2
Enter the tax rate(in percentage): 1
Bill Summary
 ----------------
price: 5000.000000
 quantity: 5
 total: 25000.000000
 discount amount: 500.000000
 tax amount: 250.000000
 final amount: 24750.000000
```