“SESSION – 5”

Q] **Consider a set of N integers. Write a C function to find the maximum and minimum integer among them along with their positions.**

ALGORITHM

STEP 1: ***Start***

STEP 2: ***Declare the Function ie sumofarray.***

STEP 3: ***Take Integer values of min, max and i.***

STEP 4: ***Run a For Loop ie (i=1;i<n;i++).***

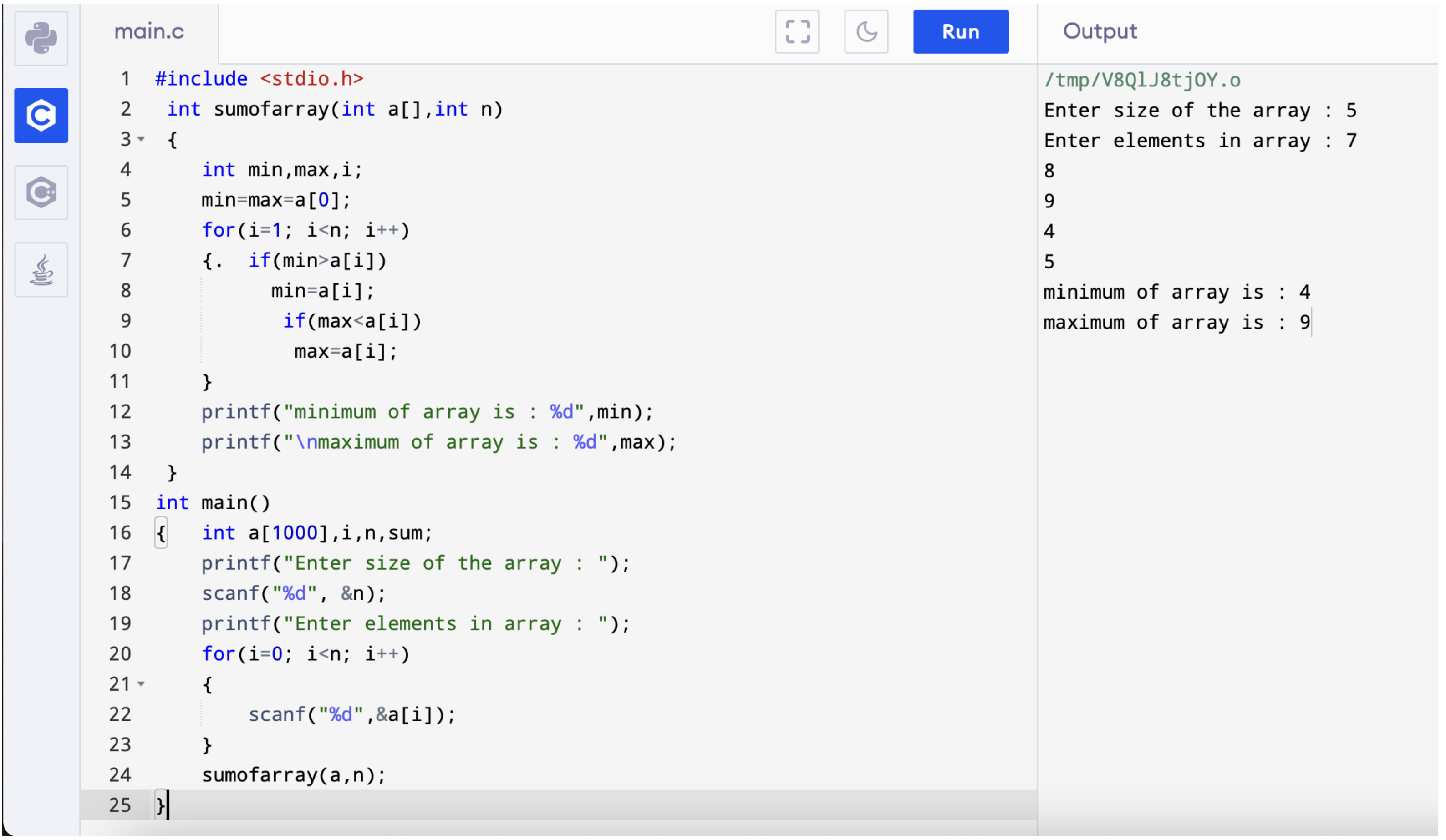
STEP 5: ***Run a If-Else statement ie if (min>a[i]) then initialize min=a[i] or if(max<a[i]) then initialize max=a[i].***

STEP 6: ***Print Minimum of array and Maximum of array.***

STEP 7: ***Print the statements ie Enter size of the array and enter elements in array.***

STEP 8: ***In the main function call the function sumofarray***

STEP 9: ***Stop.***



```c
#include <stdio.h>
 int sumofarray(int a[],int n)
 {
     int min,max,i;
     min=max=a[0];
     for(i=1; i<n; i++)
     {.  if(min>a[i])
            min=a[i];
             if(max<a[i])
              max=a[i];
     }
     printf("minimum of array is : %d",min);
     printf("\nmaximum of array is : %d",max);
 }
int main()
{    int a[1000],i,n,sum;
     printf("Enter size of the array : ");
     scanf("%d", &n);
     printf("Enter elements in array : ");
     for(i=0; i<n; i++)
     {
         scanf("%d",&a[i]);
     }
     sumofarray(a,n);
}
```

```
/tmp/V8Q1J8tjOY.o
Enter size of the array : 5
Enter elements in array : 7
8
9
4
5
minimum of array is : 4
maximum of array is : 9
```

Q] ***Write a C program to calculate the weighted average of a list of 'n' numbers, using the formula: avg= f1x1+f2x2+…+fnxn***

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***Take Integer values of n, i.***

STEP 3: ***Take Float values of num[100], sum to 0.0 & avg.***

STEP 4: ***Print the statement Enter the numbers of elements.***

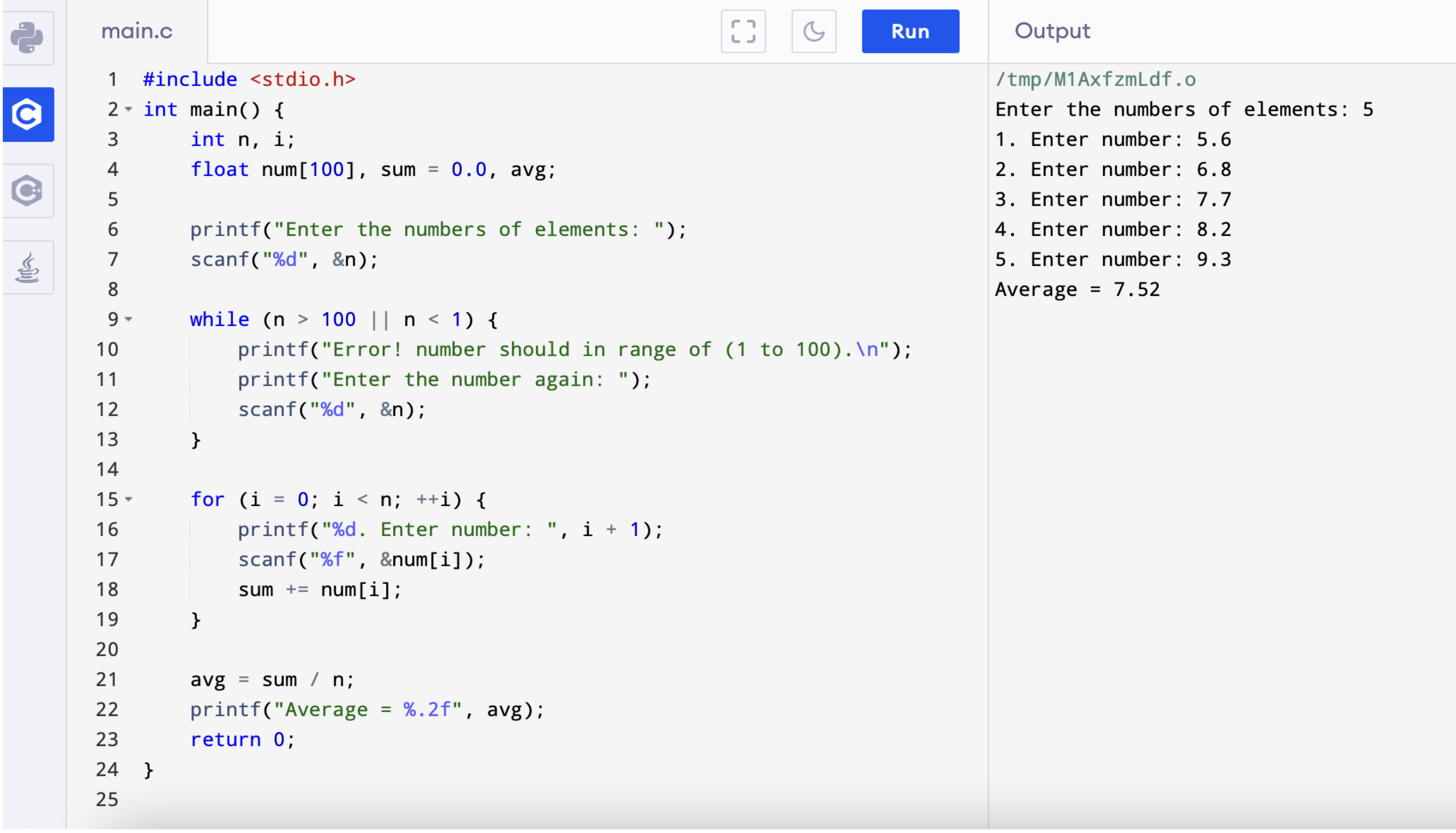
STEP 5: ***Run a While (n>100||n<1) print the statements Error! And enter the number again.***

STEP 6: ***Run a For Loop (i=0;i<n;i++) and print the statement Enter the number and print i+1.***

STEP 7: ***Initialize sum+=num[i].***

STEP 8: ***Calculate the average =sum/n and print avg.***

STEP 9: ***Stop.***



```c
#include <stdio.h>
int main() {
    int n, i;
    float num[100], sum = 0.0, avg;

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

    while (n > 100 || n < 1) {
        printf("Error! number should in range of (1 to 100).\n");
        printf("Enter the number again: ");
        scanf("%d", &n);
    }

    for (i = 0; i < n; ++i) {
        printf("%d. Enter number: ", i + 1);
        scanf("%f", &num[i]);
        sum += num[i];
    }

    avg = sum / n;
    printf("Average = %.2f", avg);
    return 0;
}
```

```
/tmp/M1AxfzmLdf.o
Enter the numbers of elements: 5
1. Enter number: 5.6
2. Enter number: 6.8
3. Enter number: 7.7
4. Enter number: 8.2
5. Enter number: 9.3
Average = 7.52
```

Q]Write a C program to create a 1dimensional array, input number from user and store in array.
Print all numbers along with the occurrence.

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***DEFINE AL to 5 as well as MAX to 5.***

STEP 3: ***Under the main function take float value of N[AL]***

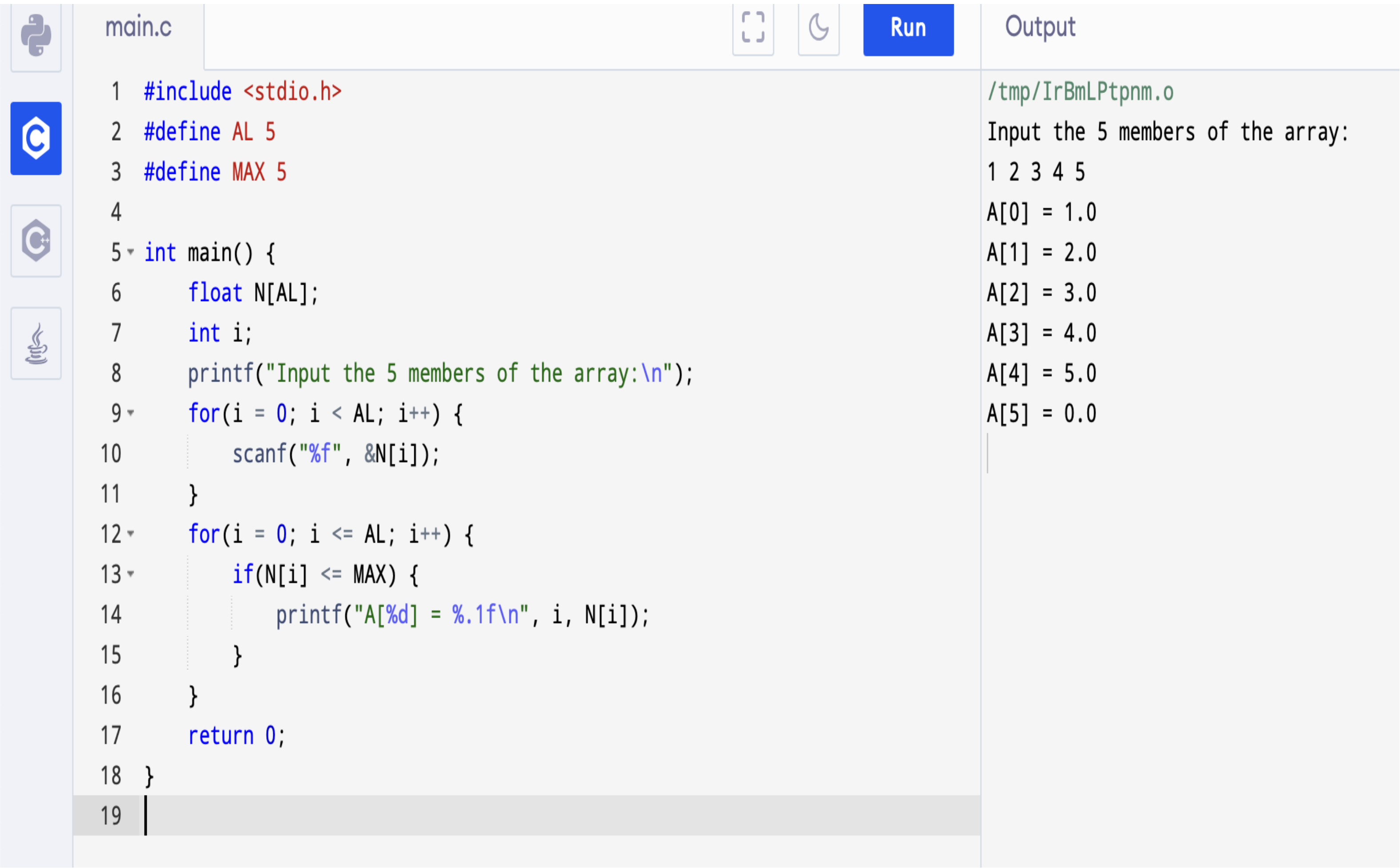
STEP 4: ***Print the statement Input the 5 members of the array***

STEP 5: ***Run a For Loop (i=0;i<AL;i++) and scanf (&N[i]).***

STEP 6: ***Run again a For Loop (i=0;i<=AL;i++) and if N[i] is less than equal to MAX .***

STEP 7: ***Then Print i and N[i] and return 0.***

STEP 8: ***Stop.***



```c
#include <stdio.h>
#define AL 5
#define MAX 5

int main() {
    float N[AL];
    int i;
    printf("Input the 5 members of the array:\n");
    for(i = 0; i < AL; i++) {
        scanf("%f", &N[i]);
    }
    for(i = 0; i <= AL; i++) {
        if(N[i] <= MAX) {
            printf("A[%d] = %.1f\n", i, N[i]);
        }
    }
    return 0;
}
```

```
/tmp/IrBmLPtpnm.o
Input the 5 members of the array:
1 2 3 4 5
A[0] = 1.0
A[1] = 2.0
A[2] = 3.0
A[3] = 4.0
A[4] = 5.0
A[5] = 0.0
```

Q]Write a C program to remove all duplicate elements in a one dimensional.
Array.

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***In the main function, take integer values of arr[10],i,j,k and Size.***

STEP 3: ***Print the statement Enter number of elements.***

STEP 4: ***Run a For Loop, (i=0;i<Size;i++) and (j=i+1; j<Size; j++) and then run a if-else statement.***

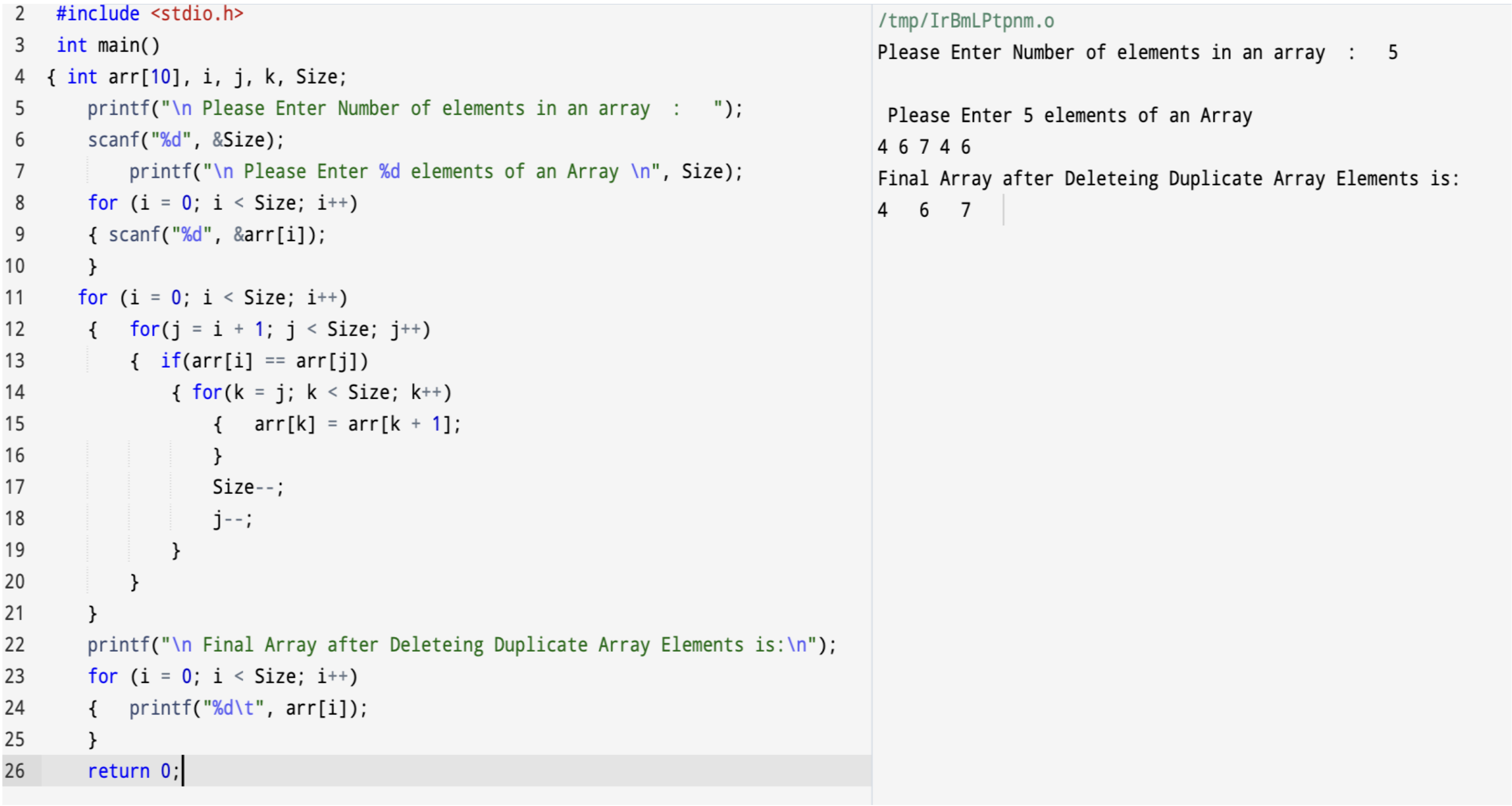
STEP 5: ***if(arr[i] == arr[j]) then arr[k] = arr[k+1].***

STEP 6: ***Decreament Size—and j--.***

STEP 7: ***Print the statement Final Array after Deleting Dublicate Array Elements.***

STEP 8: ***Print arr[i].***

STEP 9: ***Stop.***



```c
 #include <stdio.h>
 int main()
{ int arr[10], i, j, k, Size;
    printf("\n Please Enter Number of elements in an array  :   ");
    scanf("%d", &Size);
        printf("\n Please Enter %d elements of an Array \n", Size);
    for (i = 0; i < Size; i++)
    { scanf("%d", &arr[i]);
    }
   for (i = 0; i < Size; i++)
    {   for(j = i + 1; j < Size; j++)
        {  if(arr[i] == arr[j])
            { for(k = j; k < Size; k++)
                {   arr[k] = arr[k + 1];
                }
                Size--;
                j--;
            }
        }
    }
    printf("\n Final Array after Deleteing Duplicate Array Elements is:\n");
    for (i = 0; i < Size; i++)
    {   printf("%d\t", arr[i]);
    }
    return 0;
```

```
/tmp/IrBmLPtpnm.o
Please Enter Number of elements in an array  :   5

 Please Enter 5 elements of an Array
4 6 7 4 6
Final Array after Deleteing Duplicate Array Elements is:
4   6   7
```

Q]WACP to multiply two matrices and then find th e Transpose of the resultant matrix: (a) without pointer.

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***In the main function, take integer values of a[][], transpose[][], r,c.***

STEP 3: ***Print the statement Enter rows and columns.***

STEP 4: ***Run a For Loop, (i=0;i<r;i++) and (j=0; j<c; j++) thereby assigning elements to the matrix.***

STEP 5: ***Now, Printing The Matrix a[][].***

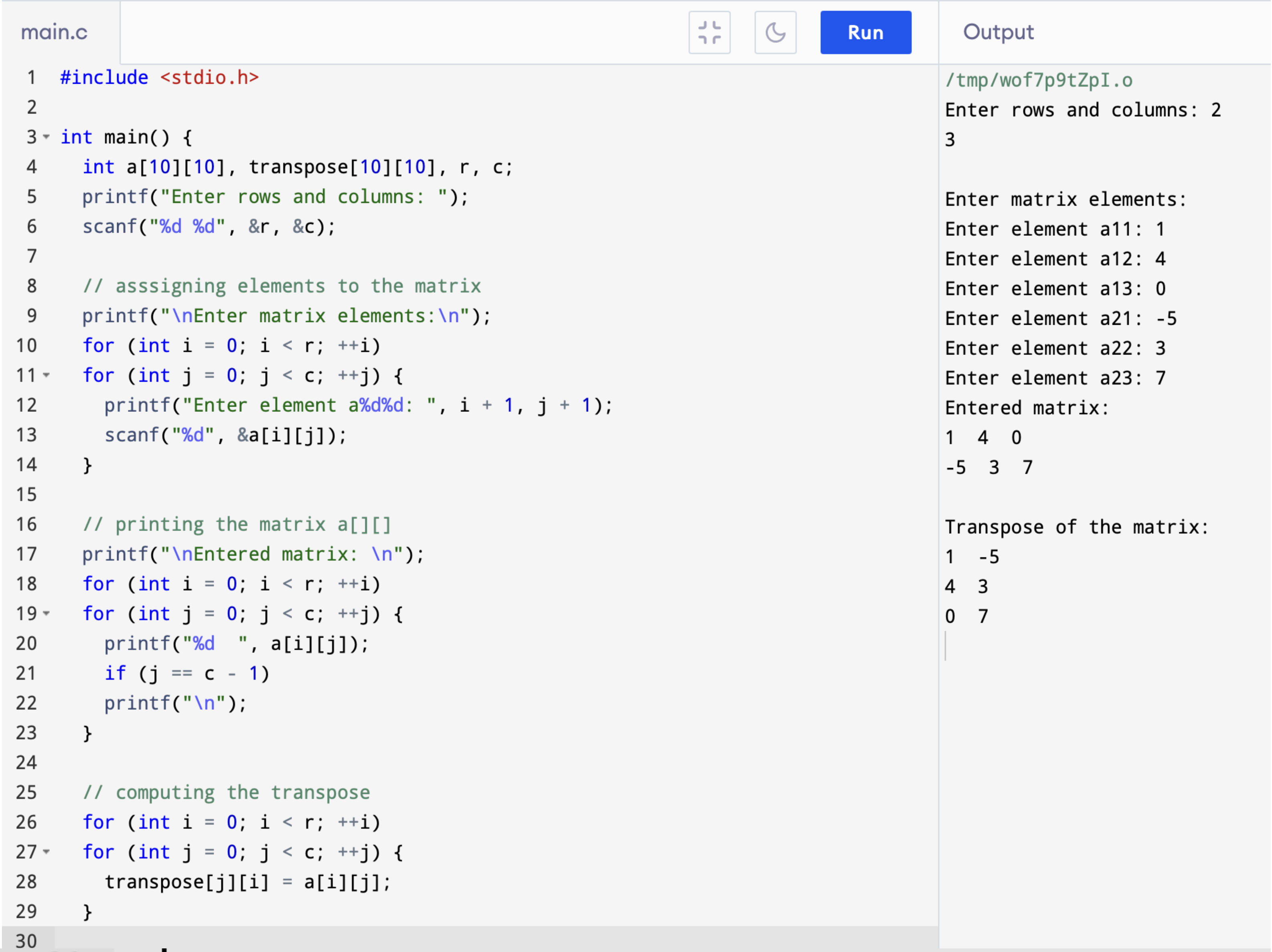
STEP 6: ***Now, Compute the Transpose. Ie tanspose[j][i]= a[i][j].***

STEP 7: ***Print the Transpose of the Matrix.***

STEP 8: ***Run Two For Loops ie (i=0; i<c; i++) and (j=0;j<r;j++).***

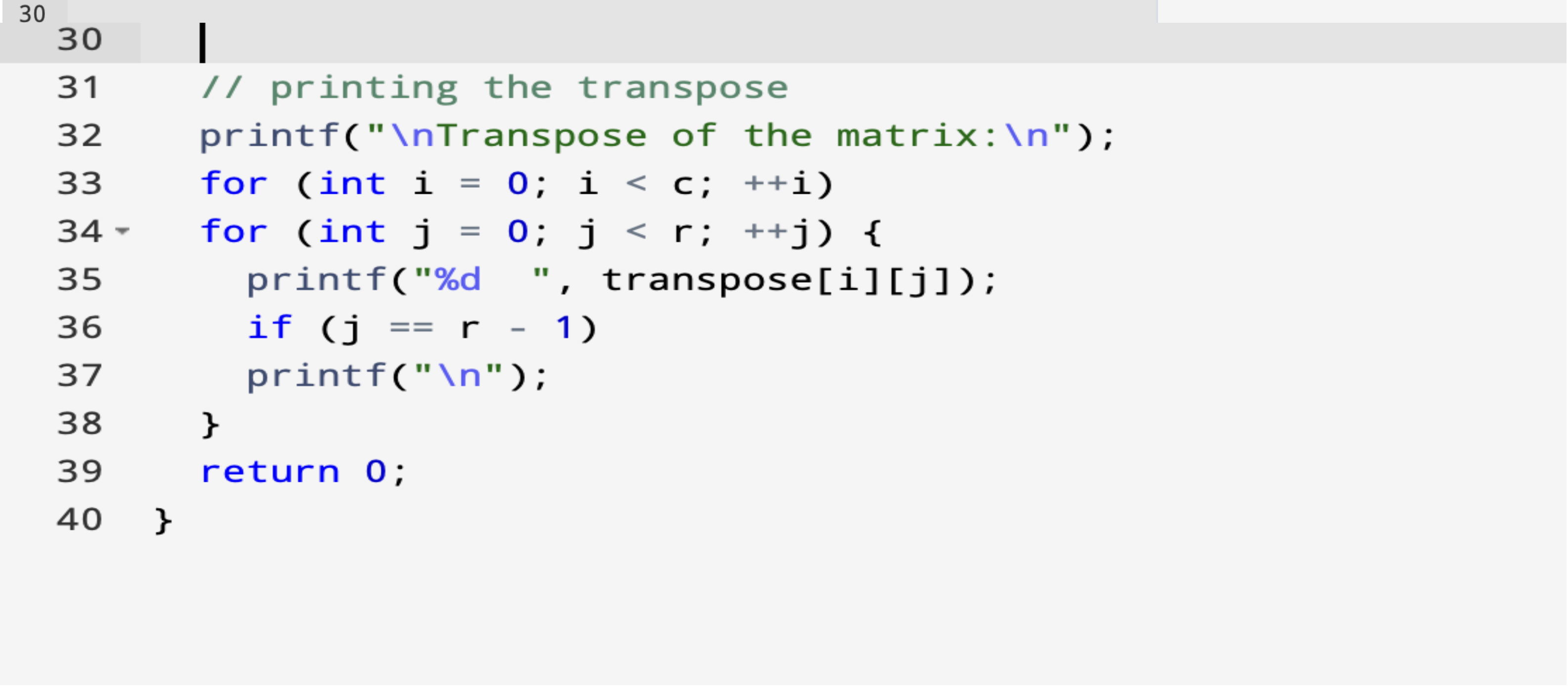
STEP 9: ***Print transpose[i][j] and check if (j==r-1).***

STEP 10 : ***Stop.***



```c
#include <stdio.h>

int main() {
  int a[10][10], transpose[10][10], r, c;
  printf("Enter rows and columns: ");
  scanf("%d %d", &r, &c);

  // asssigning elements to the matrix
  printf("\nEnter matrix elements:\n");
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    printf("Enter element a%d%d: ", i + 1, j + 1);
    scanf("%d", &a[i][j]);
  }

  // printing the matrix a[][]
  printf("\nEntered matrix: \n");
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    printf("%d  ", a[i][j]);
    if (j == c - 1)
    printf("\n");
  }

  // computing the transpose
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    transpose[j][i] = a[i][j];
  }
```

```
/tmp/wof7p9tZpI.o
Enter rows and columns: 2
3

Enter matrix elements:
Enter element a11: 1
Enter element a12: 4
Enter element a13: 0
Enter element a21: -5
Enter element a22: 3
Enter element a23: 7
Entered matrix:
1   4   0
-5   3   7

Transpose of the matrix:
1   -5
4   3
0   7
```



```c

  // printing the transpose
  printf("\nTranspose of the matrix:\n");
  for (int i = 0; i < c; ++i)
  for (int j = 0; j < r; ++j) {
    printf("%d  ", transpose[i][j]);
    if (j == r - 1)
    printf("\n");
  }
  return 0;
}
```

Q] ***Write a C Program to Print The Inverse Of A Matrix.***

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***In the main function, take integer values of mat[3][3], i and j and take float for determinant.***

STEP 3: ***Print the statement Enter elements of matrix row wise .***

STEP 4: ***Run two For Loops i and j .***

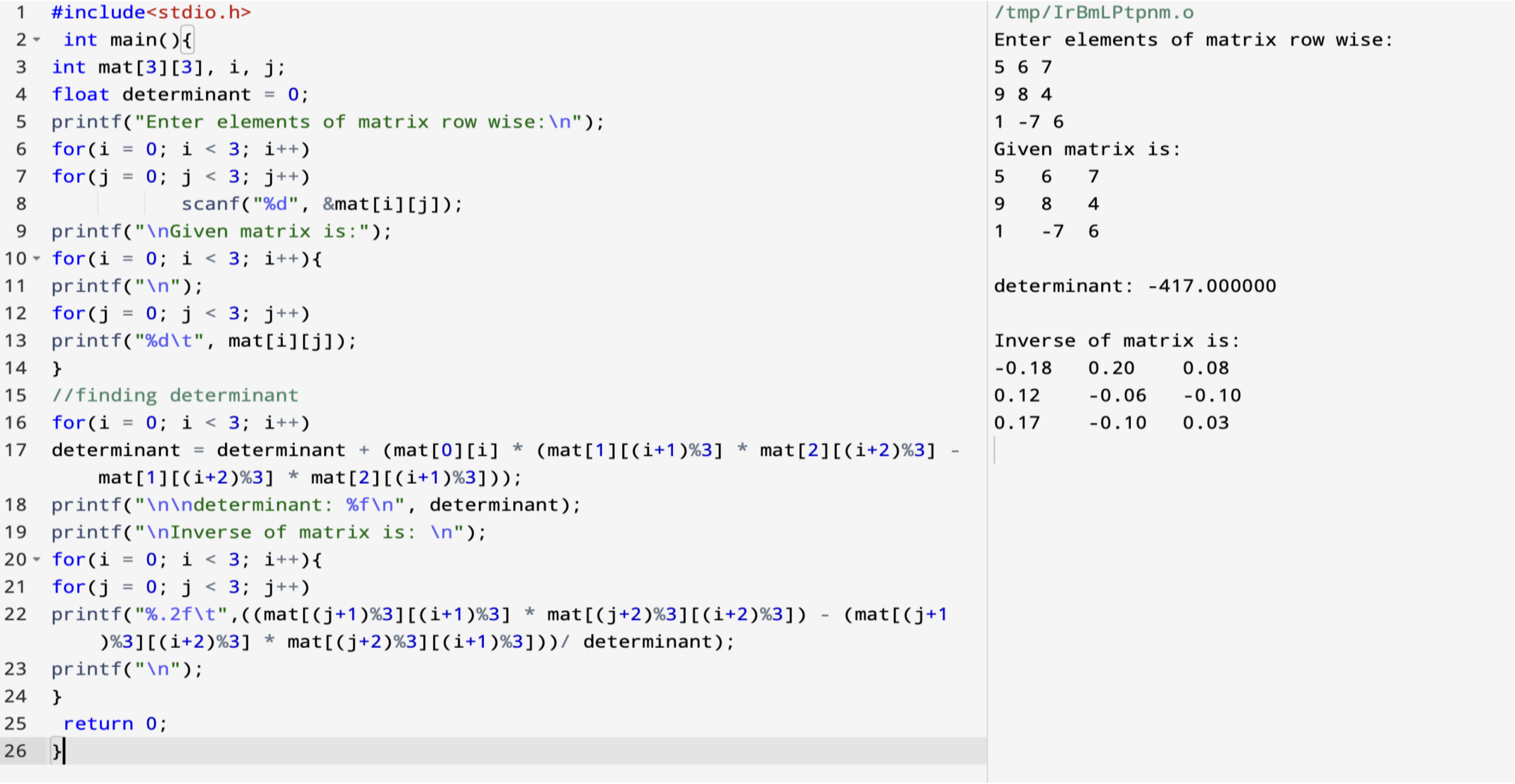
STEP 5: ***Now, Printing The statement Given Matrix is .***

STEP 6: ***Now, Print mat[i][j].***

STEP 7: ***Now, finding the Determinant and print the value of determinant after calculating it.***

STEP 8: ***Again Run the two For Loops and print mat/determinant.***

STEP 9: ***Stop.***



```c
#include<stdio.h>
 int main(){
int mat[3][3], i, j;
float determinant = 0;
printf("Enter elements of matrix row wise:\n");
for(i = 0; i < 3; i++)
for(j = 0; j < 3; j++)
          scanf("%d", &mat[i][j]);
printf("\nGiven matrix is:");
for(i = 0; i < 3; i++){
printf("\n");
for(j = 0; j < 3; j++)
printf("%d\t", mat[i][j]);
}
//finding determinant
for(i = 0; i < 3; i++)
determinant = determinant + (mat[0][i] * (mat[1][(i+1)%3] * mat[2][(i+2)%3] -
    mat[1][(i+2)%3] * mat[2][(i+1)%3]));
printf("\n\ndeterminant: %f\n", determinant);
printf("\nInverse of matrix is: \n");
for(i = 0; i < 3; i++){
for(j = 0; j < 3; j++)
printf("%.2f\t",((mat[(j+1)%3][(i+1)%3] * mat[(j+2)%3][(i+2)%3]) - (mat[(j+1
    )%3][(i+2)%3] * mat[(j+2)%3][(i+1)%3]))/ determinant);
printf("\n");
}
 return 0;
}
```

```
/tmp/IrBmLPtpnm.o
Enter elements of matrix row wise:
5 6 7
9 8 4
1 -7 6
Given matrix is:
5    6    7
9    8    4
1    -7   6

determinant: -417.000000

Inverse of matrix is:
-0.18    0.20     0.08
0.12     -0.06    -0.10
0.17     -0.10    0.03
```

Q] *Write a C program To store Names of some Authorized Persons of an organization. Develop a feature to control access of unauthorized person with their name.*

ALGORITHM

STEP 1: ***Start.***

STEP 2: ***In the main function, take integer values of I,j,n and char c[10][20] and temp[20].***

STEP 3: ***Print the statement Enter number of strings as well as Print Enter the Strings.***

STEP 4: ***Run a For Loop (i=0;i<=n;i++) and display gets(c[i]) and puts(c[i]).***

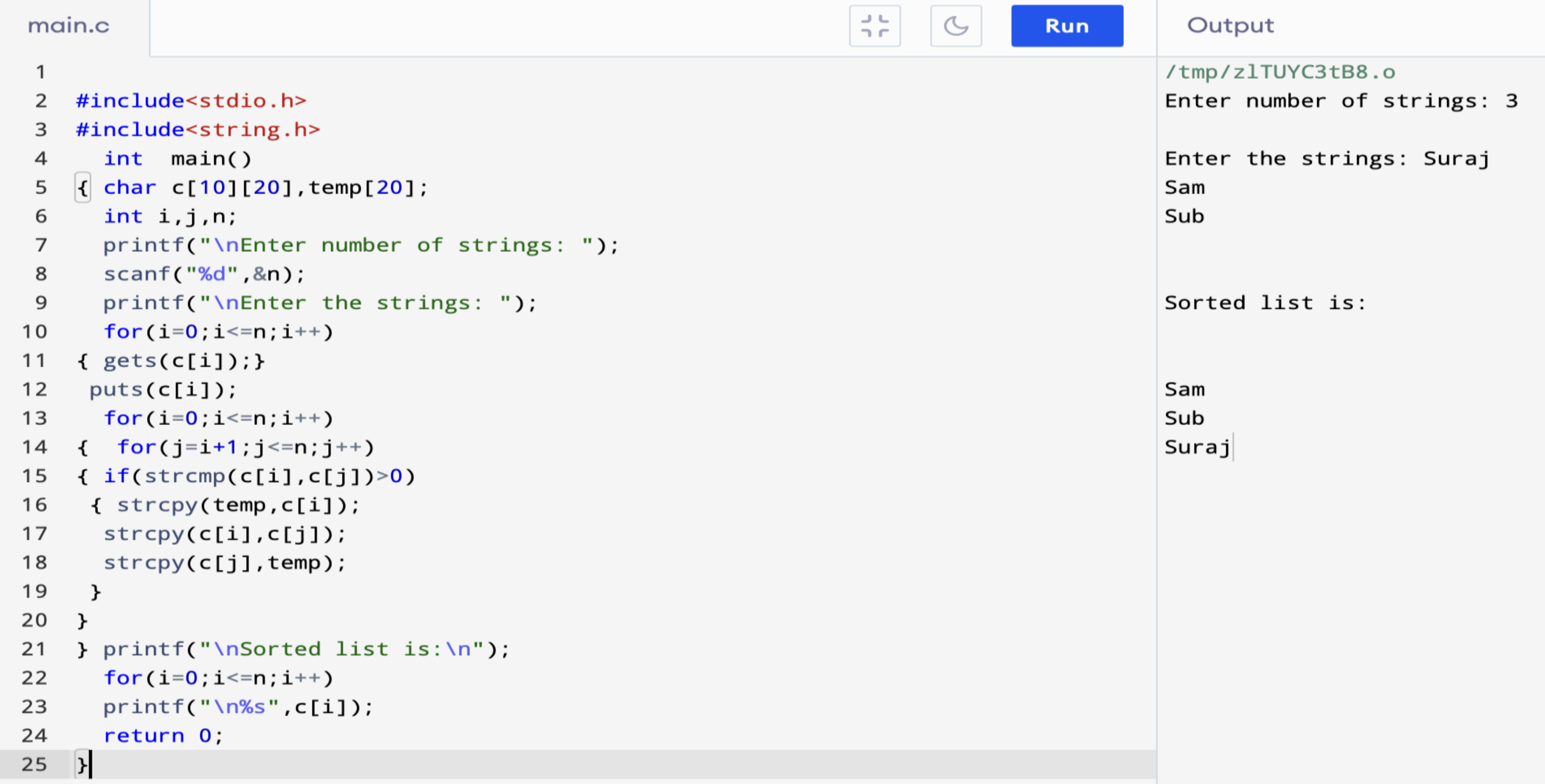
STEP 5: ***If (strcmp(c[i],c[j])>0) then display strcpy(temp,c[i]) , strcpy(c[i],c[j]) and strcpy(c[j],temp).***

STEP 6: ***Now, Print the Statement Sorted List is .***

STEP 7: ***Now, run a For Loop (i=0;i<=n;i++)***

STEP 8: ***Print c[i].***

STEP 9: ***Stop.***



```c

#include<stdio.h>
#include<string.h>
  int  main()
{ char c[10][20],temp[20];
  int i,j,n;
  printf("\nEnter number of strings: ");
  scanf("%d",&n);
  printf("\nEnter the strings: ");
  for(i=0;i<=n;i++)
{ gets(c[i]);}
 puts(c[i]);
  for(i=0;i<=n;i++)
{  for(j=i+1;j<=n;j++)
{ if(strcmp(c[i],c[j])>0)
 { strcpy(temp,c[i]);
  strcpy(c[i],c[j]);
  strcpy(c[j],temp);
 }
}
} printf("\nSorted list is:\n");
  for(i=0;i<=n;i++)
  printf("\n%s",c[i]);
  return 0;
}
```

```
/tmp/zlTUYC3tB8.o
Enter number of strings: 3

Enter the strings: Suraj
Sam
Sub


Sorted list is:


Sam
Sub
Suraj
```

THANK YOU