DATA STRUCTURE AND ALGORITHM (CSE2003)

DIGITAL ASSIGNMENT

NAME: ALOK SINHA

REG. NO.: 17BCE2380

SLOT: G2

ARRAY OF AVERAGES

Design an efficient algorithm that achieves the following task: Given an array A[1::n] of floating point numbers, it returns a two-dimensional array, say M, of size n x n in which the entry M[i][j] for i ≤j contains the average of the array entries A[i] through A[j].
That is: if i ≤ j, then
M[i][j] = A[i] +…… + A[j] / j - i + 1
whereas for i > j we have that M[i][j] = 0.
a. Describe your idea for an algorithm that creates this matrix.
b. Write down the algorithm in pseudocode.

ALGORITHM

- Start
- Take an input for the number of elements from the user.
- For the particular number of elements assigned it to the two-dimensional array as a[n][n] where n represents number of elements.
- The elements are entered at only one condition iff [i==j].
- If the condition [i > j] we make it two-dimensional as 0.
- When the condition is [i < j] we perform average operation just by adding the from i to j elements and then divide it my total number of elements taken at a time and represents in an array.
- Then the obtained results are stored in the matrix from in floating point representation with 2 value after point.
- End

PSEUDO CODE

```
Read an element n.
Write it into matrix form
if(i==j) scanf("%d",a[n][n]);
if(i>j)
set a[n][n] =0;
set i,j to 0;
set k to i;
while (i<n && i<j)
        set a[i][j] to 0;
            a[i][j] +=a[k][k];
         a[i][j]/=(j-i+1);
set i, j, k to i+1, j+1, k+1
break
set a[i][j] to 0;
run the loop
write it into matrix form a[i][j];
```

CODE

```
#include<stdio.h>
int main()
{
printf("Enter No of elements\n");
int n;
scanf("%d",&n);
float a[n][n];
int i,j,k;
printf("Enter all the elements\n");
for (i=0;i<n;i++)
{
    for (j=0;j<n;j++)
    {
        if (i==j){
            scanf("%f",&a[i][j]);
       }
    }
}
for (i=0;i<n;i++)
{
    for (j=0;j<n;j++)
    {
        if (i>j){
            a[i][j]=0;
        }
    }
```

```
}
for (i=0;i<n;i++)
{
    for (j=0;j<n;j++)
    {
        if (i<j){
                a[i][j]=0;
                for(k=i;k<=j;k++)
                {
                    a[i][j]+=a[k][k];


                }
                a[i][j]/=(j-i+1);
        }
    }
}
printf("the required matrix \n\n");
for (i=0;i<n;i++)
{
    for (j=0;j<n;j++)
    {
        printf(" %.2f ",a[i][j]);
    }
    printf("\n");
}
return 0;
}
```

"F:\c programs\ALOK SINHA\DSA\DSA_DA.exe"

```
Enter No of elements
3
Enter all the elements
1 2 3
the required matrix

 1.00  1.50  2.00
 0.00  2.00  2.50
 0.00  0.00  3.00

Process returned 0 (0x0)   execution time : 3.263 s
Press any key to continue.
```

"F:\c programs\ALOK SINHA\DSA\DSA_DA.exe"

```
Enter No of elements
4
Enter all the elements
1 2 3 4
the required matrix

 1.00  1.50  2.00  2.50
 0.00  2.00  2.50  3.00
 0.00  0.00  3.00  3.50
 0.00  0.00  0.00  4.00

Process returned 0 (0x0)   execution time : 5.622 s
Press any key to continue.
```