

# CMPE 108 - Experiment 6

## Arrays

### OBJECTIVES:

- Understand how to edit, compile and execute C computer codes.
- Understand C programming: 1-D and 2-D arrays

### NOTES:

- You should prepare the preliminary work before coming to the laboratory session and bring soft or hard copies of the preliminary work with you.
- Before writing a computer code, you should do the following steps:
  - 1) understand and analyze the problem,
  - 2) develop an algorithm and/or flowchart,
  - 3) convert the algorithm and/or the flowchart into a C code.

### PRELIMINARY WORK:

1. Show the contents of the following array declarations.

- a) `int N1[5], N2[5];`  
`N1[2] = 4;`  
`N1[0] = N1[2];`  
`N2[4] = N1[0];`
- b) `float prices[5]={1.25, 3.50, 1.20, 2.99, 0.75};`
- c) `float prices[5]={1.25, 3.50};`
- d) `int months[ ]={1,2,3,4,5,6,7,8,9,10,11,12};`
- e) `int x[2][3]={1,2,3,3,2,1};`
- f) `int y[2][3]={{1,2,3},{3,2,1}};`

2. What is the output of the following C program fragments?

- a) 

```
#include <stdio.h>
void main()
{
    int a[3][3] = {1,3,5,7,9};
    int i=0,j,sum=0;

    for(j=0; j < 3; j++)
        sum += a[1][j];
    printf("%d\n",sum);

    for(j=0; j < 3; j++)
        sum += a[j][i++];
    printf("%d\n",sum);
}
```
- b)

```
int age[4];
age[0]=2;
age[1]=43;
age[2]=60;
age[3]=46;
```

```
printf("%d\n", age[0]);
printf("%d\n", age[1]);
printf("%d\n", age[2]);
printf("%d\n", age[3]);
```

**c)**

```
int age[4] = {23, 34, 65, 74};
int same_age[4];
int i;
for(i=0; i<4; i++)
    same_age[i]=age[i];
for(i=0; i<4; i++)
    printf("same_age[%d] = %d\n", i, same_age[j]);
```

**d)**

```
int i;
float sales[120];
float sum=0.0;
for (i = 0; i < 10; i++)
{
    printf("\nEnter Sales ($) : ");
    scanf("%f", &sales[i]);
}
for (i=0;i<10;i++)
    sum+=sales[i];
printf("sum of sales is %f", sum);
```

**e)**

```
int n, m;
float x[5][6];
for (n = 0; n < 5; ++n)
    for (m = 0; m < 6; ++m)
    {
        printf("Enter value for x[%d][%d]", n, m);
        scanf("%f", &x[n][m]);
    }
for (n = 0; n < 5; n++)
    for (m = 0; m < 6; m++)
        printf("\nx[%d][%d]=%f", n, m, x[n][m]);
```

## TASKS during the LAB hours

1. Trace the following C program and find and describe the output.

```
#include <stdio.h>
int main(void)
{
    int a[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
    int b[3][3] = { {10, 11, 12}, {13, 14, 15}, {16, 17, 18} };

    int c[3][3] = { {0, 0, 0}, {0, 0, 0}, {0, 0, 0} };
    /*result array initialized to zero */.
    int i, j, k;

    printf("\n\nMATRIX a\n");

    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            printf("%d\t", a[i][j]);
        printf("\n");
    }
    /* '\t' is a tab sequence*/
    /* '\n' is a new line
    sequence*/
    /* displaying the array 'a'
    in matrix fashion*/

    printf("\n\nMATRIX b\n");

    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            printf("%d\t", b[i][j]);
        printf("\n");
    }
    /* displaying the
    array 'b' in matrix
    fashion*/

    for(i=0; i<3; i++)
        for(j=0; j<3; j++)
            for(k=0; k<3; k++)
                c[i][j] += a[i][k] * b[k][j];
    /*multiplying
    matrix 'a' by
    matrix 'b'*/
    /*result goes in
    matrix 'c'*/

    printf("\n\nMATRIX c = a x b\n");

    for (i=0; i<3; i++)
    {
        for (j=0; j<3; j++)
            printf("%d\t", c[i][j]);
        printf("\n");
    }
    // displaying the result

    return 0;
}
```

2. Consider the following code that finds the sum of elements of the integer array A:

```
#include<stdio.h>
#include<math.h>
#define MAX 5
int main()
{
    int A[MAX], i, Sum=0;

    printf("Enter the element of the array A (5 elements): ");
    for (i = 0; i < MAX; i++)
        scanf("%d", &A[i]);

    Sum = 0;
    for (i = 0; i < MAX; i++)
        Sum += A[i];

    printf("The sum is %d\n", sum);

    return 0;
}
```

- a) Edit, compile and execute this code. Use the following input values for array A elements:  
**3 2 4 1 5.**
- b) Modify the given code to find the average of the array A.

**Note:** The average is computed by as sum of elements/number of elements

A sample run of the program must be as follows:

```
Program to find the average of elements of an integer array
Enter the elements of the array A: 3 2 4 1 5
The average of the array A is 3.0.
```

3) Write a code that will do

- a) Reads the elements of two matrices M1 and M2 with the size of two rows and three columns. And then finds the sum of the two matrices, i.e.,  $C=M1+M2$ . Use the following input values for M1 and M2:

$$M1 = \begin{pmatrix} 1 & -5 & 3 \\ 2 & -1 & -3 \end{pmatrix}, M2 = \begin{pmatrix} 3 & 2 & 4 \\ -2 & -3 & 7 \end{pmatrix}$$

- b) Find and print the maximum element row of the matrix C?
- c) Find and print the average of each row of the matrix C?