ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//1. Write a C program to take n number as input in an array and print them.
#include <stdio.h>

void main() {

printf("Enter the size of the array: ");
int n;
scanf("%d", &n);
int arr[n], i = 0;

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

printf("The elements are: ");
for(i = 0; i < n; ++i)
  printf(" %d ", arr[i]);
printf("\n");
}
```

```
Enter the size of the array: 5
Enter the element1 : 1
Enter the element2 : 2
Enter the element3 : 3
Enter the element4 : 4
Enter the element5 : 5
The elements are:  1  2  3  4  5

--------------------------------
Process exited after 4.976 seconds with return value 10
Press any key to continue . . .
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//2. Write a C program to insert a number in a given position in an array.

#include <stdio.h>

void main() {
  printf("Enter the size of the array: ");
  int n = 0, num = 0, pos = 0;
  scanf("%d", &n);
  int arr[n+1], i = 0;

  for(; i < n; ++i) {
    printf("Enter the element%d : ", (i+1));
    scanf("%d", &arr[i]);
  }
  printf("Enter the element to be inserted: ");
  scanf("%d", &num);
  printf("Enter it's position: ");
  //1-based indexing is followed for the variable pos.

  scanf("%d", &pos);

  i = n;
  for(; i >= pos; --i) {
    arr[i] = arr[i - 1];
  }
  arr[pos - 1] = num;
  printf("The elements are: ");
  for(i = 0; i <= n; ++i)
    printf(" %d ", arr[i]);
  printf("\n");

}
```

```
Enter the size of the array: 5
Enter the element1 : 1
Enter the element2 : 2
Enter the element3 : 3
Enter the element4 : 4
Enter the element5 : 5
Enter the element to be inserted: 6
Enter it's position: 3
The elements are:  1  2  6  3  4  5

--------------------------------
Process exited after 12.83 seconds with return value 10
Press any key to continue . . .
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//3. Write a C program to delete a number in a given position in an array.
#include <stdio.h>
void main() {
  int n = 0, pos = 0;
  printf("Enter the size of the array: ");
  scanf("%d", &n);
  int arr[n], i = 0;

  for(; i < n; ++i) {
    printf("Enter the element%d : ", (i+1));
    scanf("%d", &arr[i]);
  }
  printf("Enter the position: ");
  //1-based indexing
  scanf("%d", &pos);

  if(pos > n)
    printf("Out of range. Deletion not possible.");
  else {
    for(i = pos - 1; i < n - 1; ++i)
       arr[i] = arr[i + 1];
   printf("The new array is ");
    for(i = 0; i < n-1; ++i)
       printf(" %d ", arr[i]);
    printf("\n");
  }
}
```

```
Enter the size of the array: 5
Enter the element1 : 1
Enter the element2 : 2
Enter the element3 : 3
Enter the element4 : 4
Enter the element5 : 5
Enter the position: 3
The new array is  1  2  4  5

--------------------------------
Process exited after 6.434 seconds with return value 10
Press any key to continue . . .
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//Q4: write a C program to search a number in a given array and also print the
 position of the input number.
#include <stdio.h>
void main() {
  int n, arr[n], i = 0, num = 0;
  printf("Enter the size of the array: ");

  scanf("%d", &n);

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

  printf("Enter the element to be found out: ");

  scanf("%d", &num);

  for(i = 0; i < n; ++i) {
    if(arr[i] == num) {
       printf("Yes! the element is found out. ");
       printf("\n index = %d ", (i+1));
       break;
    }
  }
  printf("\n");
}
```

```
Enter the size of the array: 5
Enter the element1 : 23
Enter the element2 : 34
Enter the element3 : 45
Enter the element4 : 67
Enter the element5 : 89
Enter the element to be found out: 34
Yes! the element is found out.
 index = 2

--------------------------------
Process exited after 7.082 seconds with return value 10
Press any key to continue . . .
```

```
Enter the size of the array: 4
Enter the element1 : 2
Enter the element2 : 6
Enter the element3 : 8
Enter the element4 : 3
Enter the element to be found out: 5
Element not found.

--------------------------------
Process exited after 5.348 seconds with return value 10
Press any key to continue . . .
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//Q5: Write a C program to sort an array element.

#include <stdio.h>

void main() {
  int n = 0, i = 0, j = 0;
  printf("Enter the size: ");
  scanf("%d", &n);
  int arr[n];
  for(; i < n; ++i) {
     printf("Enter the element%d : ", (i+1));
     scanf("%d", &arr[i]);
  }
  for(i = 0; i < n; ++i) {
    for(j = i+1; j < n; ++j) {
       if(arr[i] > arr[j]) {
       arr[i] ^= arr[j];
       arr[j] ^= arr[i];
       arr[i] ^= arr[j];
      }
    }
 }
  printf("The array in ascending order: ");
  for(i = 0; i < n; ++i)
    printf("%d", arr[i]);

  printf("The array in descending order: ");
  for(i = n -1; i >=0; --i)
    printf("%d", arr[i]);
}
```

```
Enter the size: 5
Enter the element1 : 34
Enter the element2 : 65
Enter the element3 : 78
Enter the element4 : 94
Enter the element5 : 26
The array in ascending order:  26  34  65  78  94
The array in descending order:  94  78  65  34  26
--------------------------------
Process exited after 7.124 seconds with return value 4
Press any key to continue . . .
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//6. Write a C program to print the address of a given input.

#include <stdio.h>

int main() {
int num = 0;
printf("Enter a number: ");
scanf("%d", &num);

printf("It's address is  %p", &num);
return 0;
}
```

```
Enter a number: 56
It's address is  000000000062FE1C
--------------------------------
Process exited after 2.028 seconds with return value 0
Press any key to continue . . .
```

```c
//Q7: Write a C program to count the number of vowel and consonant in
a character array.

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

void main() {
printf("Enter the string: ");

char arr[100];
int i = 0, vowel = 0, consonant = 0;
scanf("%[^\n]", &arr);
 for(i = 0; i < strlen(arr); ++i){
    arr[i] = tolower(arr[i]);
    if(arr[i] == 'a' || arr[i] == 'e' || arr[i] == 'i' || arr[i] == 'o' || arr
[i] == 'u')
       ++vowel;
    else
       ++consonant;
 }
 printf("Vowel Count = %d", vowel);
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
 printf("\nConsonant Count = %d", consonant);
}
```

```
C:\Users\MYPC\Documents\GitHub\Sem2_C-Programming-Classes\AD\Archana Kumari_408_07.05.21_C_Assignment\Day
Enter the string: Archana
Vowel Count = 3
Consonant Count = 4
--------------------------------
Process exited after 3.916 seconds with return value 20
Press any key to continue . . .
```

```c
//8. Write a C program to take M x N matrix as input
and print the matrix properly.

#include<stdio.h>

int main()
{
    int i,j,m,n,sum = 0;

    printf("Enter no. of rows : ");
        scanf("%d", &n);
        printf("Enter no. of cols : ");
        scanf("%d",&m);
        int mat[n][m];
        printf("Enter values to the matrix : \n");
        for (i = 0; i < n; i++)
        {
            for (j = 0; j < m; j++)
            {
                 printf("Enter a[%d][%d] value : ",i,j);
                 scanf("%d", &mat[i][j]);
        }
    }

    printf("\nThe given matrix is \n");
        for (i = 0; i < n; ++i)
        {
        for (j = 0; j < m; ++j)
            printf("\t%d", mat[i][j]);
        printf("\n");
    }

    return 0;
}
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
Enter no. of rows : 2
Enter no. of cols : 3
Enter values to the matrix :
Enter a[0][0] value : 1
Enter a[0][1] value : 2
Enter a[0][2] value : 3
Enter a[1][0] value : 4
Enter a[1][1] value : 5
Enter a[1][2] value : 6

The given matrix is
       1       2       3
       4       5       6

------------------------------
```

```c
//9. Write a C program to perform addition ans subtraction of two matrices.
#include <stdio.h>
#define MAX  10
void input(int m[][MAX], int row,int col) {
    int i,j;
    for(i=0;i< row;i++) {
        for(j=0;j< col;j++) {
            printf("Enter element matrix[%d][%d] : ",i+1,j+1);
            scanf("%d",&m[i][j]);
        }
    }
}

void print(int m[][MAX],int row,int col) {
    int i,j;
    for(i=0; i< row; i++) {
        for(j=0;j< col;j++)
            printf("%d\t",m[i][j]);
        printf("\n");
    }
}

void main() {
    int a[MAX][MAX], b[MAX][MAX], result[MAX][MAX];
    int i,j,r1,c1,r2,c2;

    printf("Enter number of Rows of matrix a: ");
    scanf("%d",&r1);
    printf("Enter number of Cols of matrix a: ");
    scanf("%d",&c1);

    printf("\nEnter elements of matrix a: \n");
    input(a,r1,c1);
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
    printf("Enter number of Rows of matrix b: ");
    scanf("%d",&r2);
    printf("Enter number of Cols of matrix b: ");
    scanf("%d",&c2);

    printf("\nEnter elements of matrix b: \n");
    input(b,r2,c2);
    if(r1==r2 && c1==c2) {
        for(i=0;i< r1;i++) {
            for(j=0;j< c1;j++)
                result[i][j] = a[i][j] + b[i][j];
        }
        printf(" Addition matrix:\n ");
        print(result,r1,c1);

        for(i=0; i< r1; i++)  {
            for(j=0; j< c1; j++)
                result[i][j] = a[i][j] - b[i][j];
        }
        printf("\nSubtracted matrix:\n");
        print(result,r1,c1);
    }
    else
        printf("\nNumber of Rows != number of Columns.");
}
```

```
C:\Users\MYPC\Documents\GitHub\Sem2_C-Programming-Classes\AD\Archana Kumari_408_07.05.21_C_Assignment\Day10_stri
Enter number of Rows of matrix a: 2
Enter number of Cols of matrix a: 2

Enter elements of matrix a:
Enter element matrix[1][1] : 4
Enter element matrix[1][2] : 7
Enter element matrix[2][1] : 2
Enter element matrix[2][2] : 9
Enter number of Rows of matrix b: 2
Enter number of Cols of matrix b: 2

Enter elements of matrix b:
Enter element matrix[1][1] : 5
Enter element matrix[1][2] : 9
Enter element matrix[2][1] : 3
Enter element matrix[2][2] : 6
 Addition matrix:
 9      16
5       15

Subtracted matrix:
-1      -2
-1      3

--------------------------------
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//10. Write a C program to find the transpose of a matrix.

#include <stdio.h>
void main(){

   int m, n, i, j;
   printf("Enter rows: ");
   scanf("%d", &m);
   printf("Enter columns: ");
   scanf("%d", &n);

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

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

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

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
Enter rows: 2
Enter columns: 3
Enter elements of the matrix
Enter element a[0][0]1
Enter element a[0][1]2
Enter element a[0][2]3
Enter element a[1][0]4
Enter element a[1][1]5
Enter element a[1][2]6
The matrix:
1       2       3
4       5       6
Transpose of the matrix:
1       4
2       5
3       6

------------------------------
Process exited after 6.74 seconds with return value 3
```

```c
//11. Write a C program to find the multiplication of two matrices.

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

void main() {
int r1, c1, r2, c2, i, j, k;

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

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

int a[r1][c1];

for(i=0; i<r1; i++)  {
   for(k=0; k<c1; k++)  {
    printf("Enter the element[%d][%d]", i, j);
    scanf("%d", &a[i][k]);
  }
}

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

printf("Enter the number of columns in second matrix: ");
scanf("%d",&c2);
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
int b[r2][c2], c[r1][c2];

for(k=0;k<r2;k++) {
   for(j=0;j<c2;j++) {
     printf("Enter the element[%d][%d]", i, j);
     scanf("%d",&b[k][j]);
   }
 }
if (c1 == r2) {
   for(i=0;i<r1;i++)  {
        for(j=0;j<c2;j++) {
           c[i][j] = 0;
           for(k=0;k<c1;k++) {
              c[i][j] += a[i][k] * b[k][j];
            }
        }
   }
 printf("\nMultiplication Matrix:\n");
   for(i=0; i<r1 ;i++) {
       for(j=0; j < c2 ;j++)
          printf (" %d ",c[i][j]);
        printf("\n");
   }
 }
else
printf ("\nMultiplication not possible.");
}
```

```
Enter the number of rows in first matrix: 2
Enter the number of columns in first matrix: 2
Enter the element[0][1]3
Enter the element[0][1]7
Enter the element[1][1]9
Enter the element[1][1]5
Enter the number of rows in second matrix: 2
Enter the number of columns in second matrix: 2
Enter the element[2][0]1
Enter the element[2][1]0
Enter the element[2][0]0
Enter the element[2][1]1

Multiplication Matrix:
 3  7
 9  5

--------------------------------
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```c
//12. Write a C program to addition and subration of two matrices
using pointer and malloc() function.

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

void main() {

//*(*(p + i) + j) == p[i][j]

  int  **p, **q, i, j, row, col;
  printf("Enter rows(s): ");
  scanf("%d", &row);
  printf("Enter col(s): ");
  scanf("%d", &col);

  p = (int **)malloc(sizeof(int *)*row);

  for(i = 0; i < row; ++i) {
  *(p + i) = (int *)malloc(sizeof(int)*col);
  }

  for(i = 0; i <row; ++i) {
    for (j = 0; j < col; ++j) {
       printf("Enter the element[%d][%d] : ", i, j);
       scanf("%d", *(p + i) + j);
    }
  }
  printf("\n");
 q = (int **)malloc(sizeof(int *)*row);

  for(i = 0; i < row; ++i) {
  *(q + i) = (int *)malloc(sizeof(int)*col);
  }

  for(i = 0; i <row; ++i) {
    for (j = 0; j < col; ++j) {
       printf("Enter the element[%d][%d] : ", i, j);
       scanf("%d", *(q + i) + j);
    }
  }
 printf("The addition matrix is : \n");
for(i = 0; i <row; ++i) {
    for (j = 0; j < col; ++j) {
       printf("  %d  ", (*(*(p + i) + j) + *(*(q + i) + j)));
    }
    printf("\n");
```

ARCHANA KUMARI
DEPT. - ECE
ROLL NO. - 408

```
  }

 printf("The subtraction matrix is : \n");
for(i = 0; i <row; ++i) {
    for (j = 0; j < col; ++j) {
       printf("  %d  ", (*(*(p + i) + j) - *(*(q + i) + j)));
    }
    printf("\n");
  }

}
```

```
Enter rows(s): 2
Enter col(s): 2
Enter the element[0][0] : 3
Enter the element[0][1] : 6
Enter the element[1][0] : 1
Enter the element[1][1] : 9

Enter the element[0][0] : 3
Enter the element[0][1] : 5
Enter the element[1][0] : 9
Enter the element[1][1] : 5
The addition matrix is :
  6    11
  10    14
The subtraction matrix is :
  0    1
  -8    4


--------------------------------
Process exited after 17.53 seconds with return value 2
```