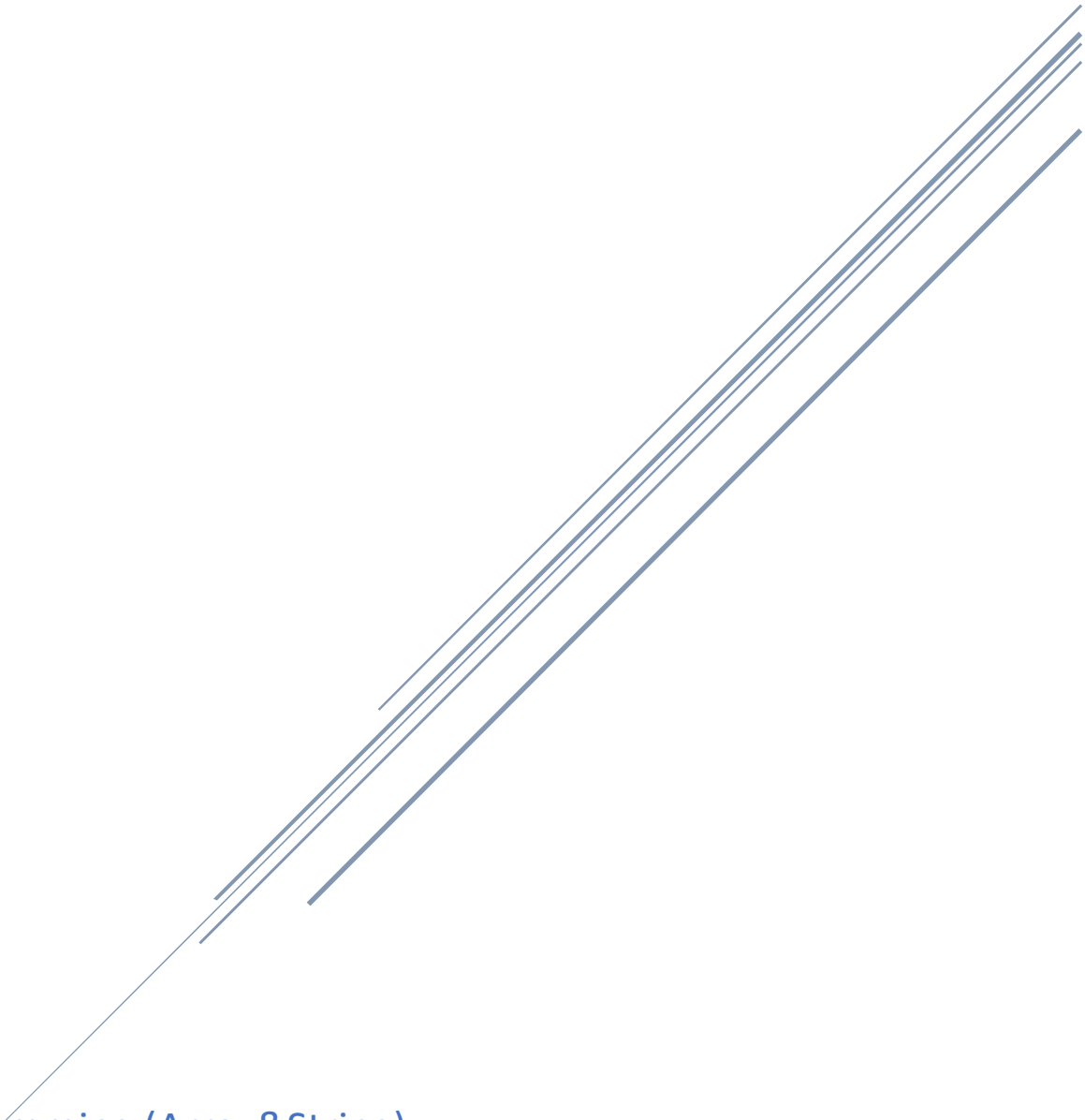
ASSIGNMENT

Lecture 4



C-Programming (Array&String)

ENG/Mostafa Hassan Aboshaker

Array

EX1: Example of Multidimensional Array In C

Write a C program to find sum of two matrix of order 2*2 using multidimensional arrays where, elements of matrix are entered by user.

```c
#include <stdio.h>
int main() {
    int matrix1[2][2], matrix2[2][2], sum[2][2];
    printf("Enter elements for the first matrix (2x2):\n");

    for (int i = 0; i < 2; ++i) {
        for (int j = 0; j < 2; ++j) {
            printf("Enter element at position %d,%d: ", i + 1, j + 1);
            scanf("%d", &matrix1[i][j]);}}
    printf("\nEnter elements for the second matrix (2x2):\n");

    for (int i = 0; i < 2; ++i) {
        for (int j = 0; j < 2; ++j) {
            printf("Enter element at position %d,%d: ", i + 1, j + 1);
            scanf("%d", &matrix2[i][j]);}}

    for (int i = 0; i < 2; ++i) {
        for (int j = 0; j < 2; ++j) {
            sum[i][j] = matrix1[i][j] + matrix2[i][j];}}

    printf("\nSum of the two matrices:\n");
    for (int i = 0; i < 2; ++i) {
        for (int j = 0; j < 2; ++j) {
            printf("%d ", sum[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

EX2: C Program to Calculate Average Using Arrays

```c
#include <stdio.h>

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

    int array[n];
    float sum = 0;

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

    double average = sum / n;

    printf("Average of the elements: %.2lf\n", average);

    return 0;
}
```

EX3: C Program to Find Transpose of a Matrix

```c
#include <stdio.h>

int main() {
    int row, col,matrix[row][col],transpose[col][row];
    printf("Enter the number of rows: ");
    scanf("%d", &row);
    printf("Enter the number of columns: ");
    scanf("%d", &col);
    if (row <= 0 || col <= 0) {
        printf("Please enter valid dimensions for the matrix.\n");
        return 1; }
    printf("Enter elements for the matrix (%dx%d):\n", row, col);
    for (int i = 0; i < row; ++i) {
        for (int j = 0; j < col; ++j) {
            printf("Enter element at position %d,%d: ", i + 1, j + 1);
            scanf("%d", &matrix[i][j]);}}

    printf("\nOriginal Matrix:\n");
    for (int i = 0; i < row; ++i) {
        for (int j = 0; j < col; ++j) {
            printf("%d\t", matrix[i][j]);
        }printf("\n");}
    for (int i = 0; i < col; ++i) {
        for (int j = 0; j < row; ++j) {
            transpose[i][j] = matrix[j][i];  }}

    printf("\nTranspose of the Matrix:\n");
    for (int i = 0; i < col; ++i) {
        for (int j = 0; j < row; ++j) {
            printf("%d\t", transpose[i][j]);
        }printf("\n");}
    return 0;
}
```

EX4: C Program to Insert an element in an Array

```
int main() {
    int n, position, element,array[100];
    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; ++i) {
        printf("Element %d: ", i + 1);
        scanf("%d", &array[i]);}
    printf("Enter the position to insert the element: ");
    scanf("%d", &position);
    if (position < 1 || position > n + 1) {
        printf("Invalid position. Please enter a position between 1 and %d.\n", n + 1);
        return 1;}
    printf("Enter the element to insert: ");
    scanf("%d", &element);

    for (int i = n; i >= position; --i) {
        array[i] = array[i - 1];}
    array[position - 1] = element;
    printf("\nArray after insertion:\n");
    for (int i = 0; i <= n; ++i) {
        printf("%d ", array[i]);
    }
    printf("\n");

    return 0;
}
```

EX5: C Program to Search an element in Array

```
#include <stdio.h>

int main() {
    int n, searchElement,array[100];
    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);

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

    printf("Enter the element to search: ");
    scanf("%d", &searchElement);

    int found = 0; // 0 represents false, 1 represents true
    int position;
    for (int i = 0; i < n; ++i) {
        if (array[i] == searchElement) {
            found = 1;
            position = i + 1;
            break;  }}

    if (found)
        printf("Element %d found at position %d.\n", searchElement, position);
     else
        printf("Element %d not found in the array.\n", searchElement);

    return 0;
}
```

String

Ex1: C Program to Find the Frequency of Characters in a String

```
#include <stdio.h>
int main(){
char c[1000],ch;
int i, count=0;
printf("Enter a string: ");
gets(c);
printf("Enter a character to find frequency: "); scanf("%c", &ch);
for(i=0;c[i]!= '\0'; ++i)
{
    if(ch==c[i])
    ++count;
}

printf("Frequency of %c = %d", ch, count);
return 0;
}
```

EX2: C Program to Find the Length of a String

```
#include <stdio.h>
int main()
{
char s[1000];
int i;
printf("Enter a string: "); scanf("%s",s);
for(i=0; s[i]!= '\0'; ++i); printf("Length of string: %d",i);
return 0;
}
```

EX3: C Program to Reverse String Without Using Library Function

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

int main() {
char str[100], temp;
int i, j = 0;
printf("\nEnter the string :");
gets(str);
i = 0;
j=strlen(str) - 1;
while (i < j) {
    temp = str[i];
    str[i] = str[j];
    str[j] = temp;
    i++;
    j--;
}


printf("\nReverse string is :%s", str);
return (0);
}
```