Programming Fundamentals (COMP1112) Arrays

Division of Science & Technology
University of Education, Lahore.

Arrays

- Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

Declaring array

- To declare an array, define the variable type, specify the name of the array followed by **square brackets** and specify the number of elements it should store.

```
dataType arrayName[arraySize];
```

Examples:

```
float mark[5];
string cars[4];
string cars[4] = {"Volvo", "BMW", "Ford", "Mazda"};
```

To create an array of three integers, you could write:

```
int myNum[3] = {10, 20, 30};
```

Access the Elements of an Array

- You access an array element by referring to the index number.
- This statement accesses the value of the **first element** in **cars**:

```
string cars[4] = {"Volvo", "BMW", "Ford", "Mazda"};
cout << cars[3];
```

Loop Through an Array

You can loop through the array elements with the `for` loop.
The following example outputs all elements in the **cars** array

```
string cars[4] = {"Volvo", "BMW", "Ford", "Mazda"};
for(int i = 0; i <4; i++) {
  cout << cars[i] << "\n";
}
```

Omit Array Size

- You don't have to specify the size of the array. It will be as big as the elements that are inserted into it:

```
string cars[] = {"Volvo", "BMW", "Ford"}; // size of array is 3
```

Example- calculating average of 10 numbers

```
int main()
{
int i;
float num[10], sum=0.0, average;
for(i = 0; i < 10; ++i)
{
cout << i + 1 << ". Enter number: ";
cin >> num[i];
sum += num[i];
}
average = sum / 10;
cout << "Average = " << average;
return 0;
}
```

Example- Finding largest element of array

```
int i, n;
    float arr[10];
    cout << "Enter total number of elements: ";
    cin >> n;
for(i = 0; i < n; ++i)
    {
       cout << "Enter Number " << i + 1 << " : ";
       cin >> arr[i];
    }
    // Loop to store largest number to arr[0]
    for(i = 1;i < n; ++i)
    { if(arr[0] < arr[i])
           arr[0] = arr[i];
    }
    cout << "Largest element = " << arr[0];
```

Example- take 5 numbers in array from user and display those in reverse order

```
int arr[5];
for(int i = 0; i < 5; ++i)
{
cout << "Enter Number " << i + 1 << " : ";
cin >> arr[i];
}
cout << "reverse order"<<endl;
for(int i = 4;i>=0; i--)
{
cout << arr[i];
}
```

Array of characters

Two ways of declaration and initialization:

- char name[]="khan ali";
- char name2[]={'a','b','c'};

Note: ‘\0’ is inserted as end of string incase of name[]

Exercise

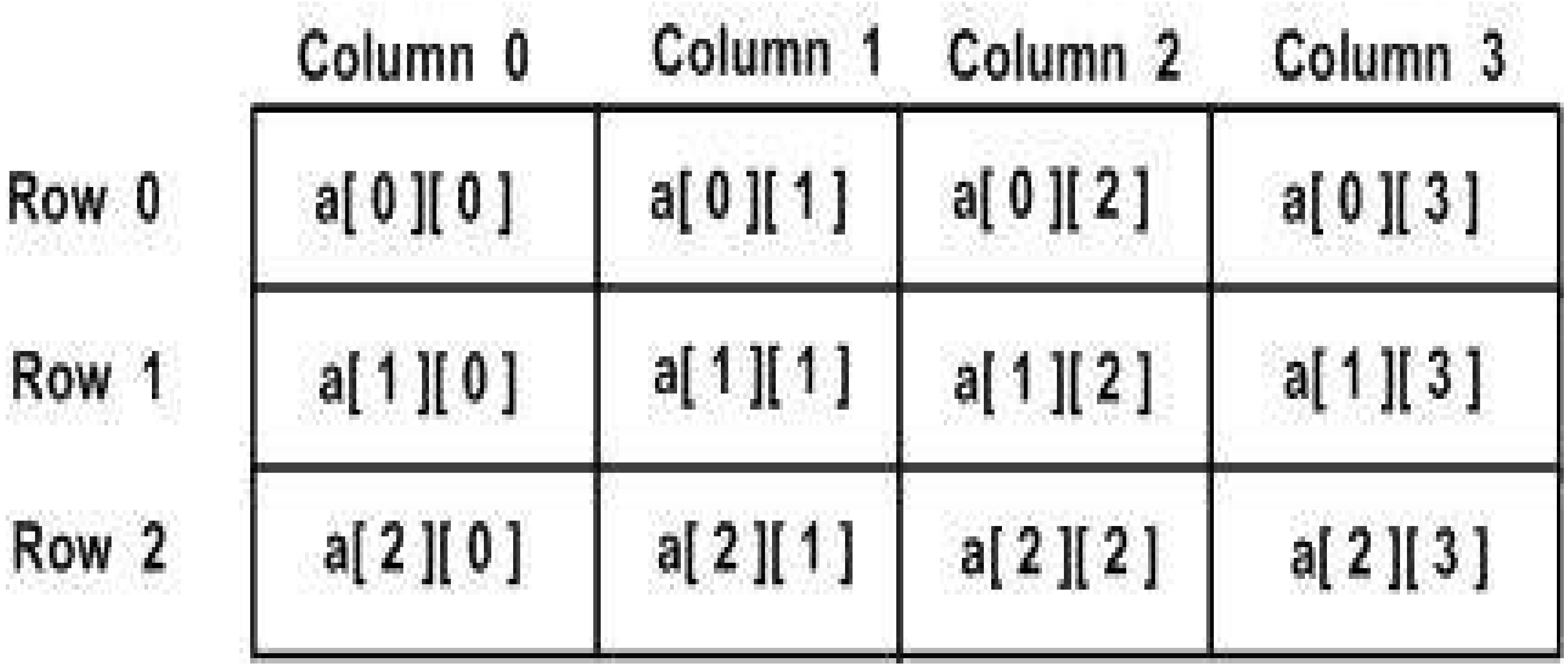
- Write a C++ program that checks palindromes (e.g. MADAM) entered by user and displays appropriate messages on finding and not finding a string as palindrome.

Two-dimensional or 2D array

- In **C++ two Dimensional array** is an **array** that consists of more than one rows and more than one column. In 2-D **array** each element is refer by **two** indexes. Elements stored in these **Arrays** in the form of matrices. The first index shows a row of the matrix and the second index shows the column of the matrix.
- A two-dimensional array is, in essence, a list of one-dimensional arrays.
- Declared as

type arrayName [R][C];

Formation



	Column 0	Column 1	Column 2	Column 3
Row 0	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Row 1	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Row 2	a[2][0]	a[2][1]	a[2][2]	a[2][3]

- Arr[0][0]=10; // 10 stored in first column of first row
- Arr[0][1]=20; // 20 stored in second column of first row
- Arr[0][2]=30; // 30 stored in third column of first row
- Arr[1][0]=40; // 40 stored in first column of second row
- Arr[1][1]=50; // 50 stored in second column of second row
- Arr[1][2]=60; // 60 stored in third column of second row

Initializing 2D Array

- The process of assigning values during declaration is called initialization. The 2D array can be initialized by putting the curly braces around each row separating by a comma also each element of a matrix should be separated by a comma.

```
int mat [3][3]= {
{ 3,6,8 },
{ 5,4,7 },
{ 2,4,7 }
};
int mat[3][3]={3, 6, 8, 5, 4, 7, 2, 4, 7};
```

```
main()
{
int k=0;
int matrix [5] [5];
for (int i=0 ; i<5 ; i++)
{
    for (int j=0 ; j<5 ; j++)
    {
     matrix [i] [j] =  ++k;
    }
}
}
```

Entering data in 2D array

- Nested loop is used to enter data in 2-D arrays. Generally, the outer loop acts as the number of rows of a matrix and the inner loop acts as the number of columns of a matrix.

```
int arr[2][4];

for( int i =0; i <2;i++)

for(int j=0;j<4;j++)

cin>>arr[i][j];
```

```
int matrix [2][3];
//Taking integer inputs in a matrix

for (int m1=0 ; m1<2 ; m1++)
{
    for (int m2=0 ; m2<3 ; m2++)
    {
     cout<<"Enter Integer :";
     cin>>matrix [m1][m2];
     }
}
cout<<endl;
```

```
//Displaying elements of a matrix

for (int m1=0 ; m1<2 ; m1++)
{
    for (int m2=0 ; m2<3 ; m2++)
    {
     cout<<"Your Entered Integer are :";
     cout<<matrix [m1][m2];
     cout<<endl;
    }
}
```

```
//Program to initialize a 2D array and display maximum and minimum numbers
int max, min;
int arr[2][4]= {{15, 21, 9, 84},{33, 72, 18, 47}};
max=min =arr[0][0];
for (int i=0;i<2;i++)
for (int j=0;j<4;j++)
{
if (arr[i][j]>max)
max = arr[i][j];
if (arr[i][j]<min)
min = arr[i][j];
}
cout<<“Maximum:  “<<max<<“  Minimum:  “<<min;
}
```

References

- C++ How to Program
  By Deitel & Deitel

- The C++ Programming Language
  By Bjarne Stroustrup

- Object oriented programming using C++ by Tasleem Mustafa, Imran Saeed, Tariq Mehmood, Ahsan Raza

- https://www.tutorialspoint.com/cplusplus

- http://ecomputernotes.com/cpp/introduction-to-oop

- http://www.cplusplus.com/doc/tutorial

- https://www.guru99.com/c-loop-statement.html

- www.w3schools.com