

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

(Week 12) Lecture 23 and 24

Objectives: Learning objectives of this lecture are

- **Char Array**
- **cin and cout on char array**
- **Primitive Types vs Non Primitive Types**
- **Programmer defined type arrays.**



Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Lecture # 23 Char Array

So far we have discussed arrays of numeric data type. In this lecture we will be discussing array of characters. When to use array of characters? Answer to the question is simple when we have to store text data (String) we shall be using char array. For example storing name of a person or storing address of a location or storing information of NIC etc. Whenever we need text data it may be alphanumeric data or alphabets we shall be using char array.

Declaration of char Array:

Declaration of char array is same as of array of integer or any other array. So let us discuss both way static and dynamic declaration of array.

Static: Syntax for declaration of static char array is given below.

char name[10];

Dynamic: Dynamic array is declared using pointer and new keyword as syntax is given below.

char *name = new char[10];

Declaration :

Static Array

char name[10];

Dynamic Array

char *name = new char [10];

0	1	2	3	4	5	6	7	8	9

Figure 1 Static and Dynamic Declaration of char Array

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Figure 1 explains declaration of char array static and dynamic as in both the cases memory of 10 characters is reserved. Static is allocated at compile time however dynamic is allocated at run time.

Initialization and Output:

Figure given below explains initialization of char array.

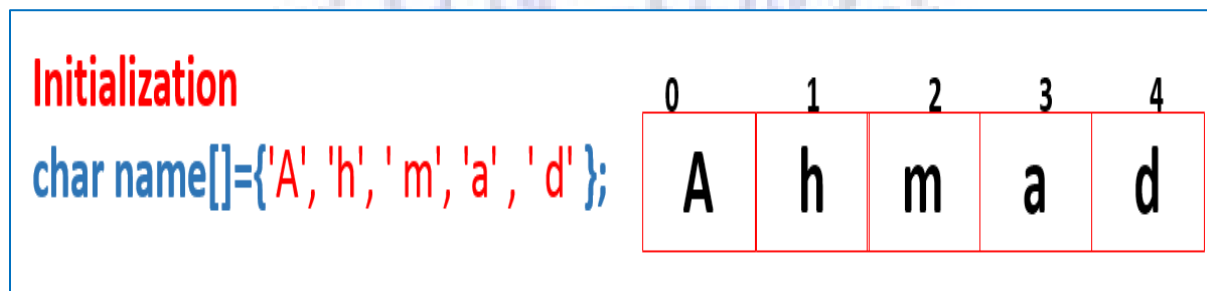


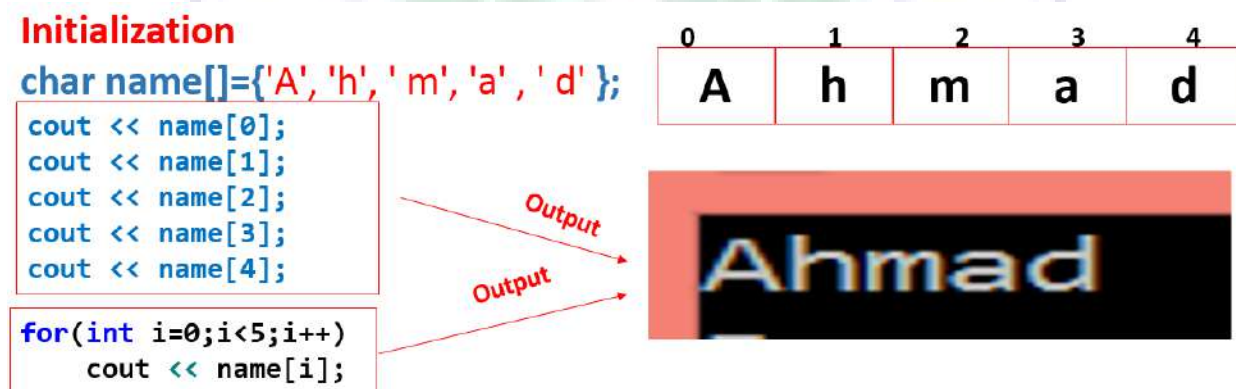
Figure 2 Initialization of char Array

Above figure explains initialization of char array. Where a char array of size 4 is created and characters are stored in a sequence.

What if we want to print data of above array it can be done in three different ways.

1. Print each index separately through cout.
2. Use loop to print value of each index.
3. Print array using cout e.g cout<<name;

Let us discuss these ways of printing data and array to be considered is the array mentioned in Figure 2.



Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

What if we print the array using cout only. Note that cout statement has polymorphic behavior in case of printing integer array and char array.

```
cout << name;
```

Output

Ahmad [garbled characters]

Note that data of array is printed however some unknown characters are also printed that were not stored in array this is because cout statement keeps on printing data until null character is not found.

What if we change the initialization of our array as follow.

Initialization

```
char name[]={ 'A','h','m','a','d','\0' };
```

```
cout << name[0];  
cout << name[1];  
cout << name[2];  
cout << name[3];  
cout << name[4];  
cout << name[5];
```

```
for(int i=0;i<6;i++)  
    cout << name[i];
```

```
cout << name;
```

	0	1	2	3	4	5
	A	h	m	a	d	\0

Ahmad

Dec	Hex	Oct	Char
0	0	000	NUL (null)
1	1	001	SOH (start of heading)
2	2	002	STX (start of text)
3	3	003	ETX (end of text)
4	4	004	EOT (end of transmission)
5	5	005	ENQ (enquiry)
6	6	006	ACK (acknowledge)
7	7	007	BEL (bell)
8	8	010	BS (backspace)
9	9	011	TAB (horizontal tab)
10	A	012	LF (NL line feed, new line)
11	B	013	VT (vertical tab)
12	C	014	FF (NP form feed, new page)
13	D	015	CR (carriage return)
14	E	016	SO (shift out)

Output of above statements is same. In above case cout << name; prints Ahmad only this is because cout statement prints entire character array up to first null character. Note that '\0' is same as 0 and NULL as shown in initialization given below.

```
char name[]={ 'A','h','m','a','d','\0' };
```

```
char name[]={ 'A','h','m','a','d',0 };
```

```
char name[]={ 'A','h','m','a','d',NULL };
```

'\0' = 0 = NULL in case of character all have same meaning that is null character.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar email id: athar@northern.edu.pk, Whatsapp# 0333-5077664
Now what if we place multiple null character in array as given below.

```
char name[]={ 'A','h','m','\0','d' , '\0' };
```

0	1	2	3	4	5
A	h	m	\0	d	\0

Now what if we print the above array as with using different statement.

```
for(int i=0;i<6;i++)  
    cout << name[i];
```

output

Ahm d

```
for(int i=0;name[i]!='\0';i++)  
    cout << name[i];
```

output

Ahm

output

```
cout << name;
```

Input in char Array:

Consider following code given below for input in array.

```
char name[6];
```

```
cout << "Enter name ";
```

```
for (int i = 0; i<6; i++)
```

```
    cin >> name[i];
```

There is one problem in above code what if characters of name is 3 i.e we want to store Ali in array above code is restricting us to provide input for every index, we may not be storing data in all indexes.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

0	1	2	3	4	5
A	L	I	#	?	\$

We Just wanted to store Ali in array but above code enforced us to provide values for 3,4 and 5 index. We shall be using just cin statement for taking input in case of char array as follow.

cin>>name; // note that if name is integer type array than it is error.

```
char name[6];  
cout << "Enter name ";  
cin >> name;
```

0	1	2	3	4	5
A	L	I	\0		

Enter name A L I

Note that cin statement work on key press when on console window A is pressed it is moved to 0 index after that L is pressed it is moved to 1 index than I is pressed is moved to 2 index after that enter is pressed \0 is placed at 3rd index. Cin on char array works on key press this is because each index stores on a single character.

```
char name[10];  
cout << "Enter a name ";  
cin >> name; // Ali is entered  
cout << "You entered ";  
cout << name; // Display data upto first null character is appeared.
```

cout << name; can be replaced by loop given below.

```
int i = 0;  
while (name[i] != '\0')  
{  
    cout << name[i];  
    i++;  
}
```

As the above loop also prints all the character up to first null character.

cin vs gets/gets_s

While taking input using cin statement there is a problem cin statement does not handles spaces.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
char name[10];  
cout << "Enter name ";  
cin >> name;
```

0	1	2	3	4	5	6	7	8	9
B	S	C	S	\0					

Value entered

Enter name BSCS 2A

If value entered is BSCS 2A. Space after BSCS is considered as terminator in case of char array and rest of characters are ignore. If we want to store data with spaces than use gets/gets_s.

```
char name[10];  
cout << "Enter name ";  
gets_s(name);
```

0	1	2	3	4	5	6	7	8	9
B	S	C	S		2	A	\0		

Enter name BSCS 2A

Now in some editors you wont find gets_s than use gets with same syntax of gets_s. The difference between cin and gets/gets_s is that cin does not handles spaces however gtes handles all data with spaces.

Output of character

For displaying data of character array follow one of following way.

Way 1	Way 2	Way 3
<pre>cout << name;</pre> Will display entire array up to first null character	<pre>int i = 0; while (name[i] != '\0') { cout << name[i]; i++; }</pre> Loop through all the indexes until a null character is appeared.	<pre>puts(name);</pre> Also displays entire char array.

Program 1: Input in char array of size 10 and display length of array.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Solution: What does length of array means? It actually means number of characters up to first null character.

```
#include "iostream"
using namespace std;
void main()
{
    char arr[10];
    cout << "Enter data ";
    cin >> arr;
    int i = 0;
    while (arr[i] != '\0')
        i++;
    cout << "Length " << i << endl;
}
```

Assume tahat array Contains

0	1	2	3	4	5	6	7	8	9
A	L	I	\0						

i=0 → arr[i] != '\0' → arr[0] != '\0' → A != '\0' → True → In body increment i → i=1

i=1 → arr[i] != '\0' → arr[1] != '\0' → L != '\0' → True → In body increment i → i=2

i=2 → arr[i] != '\0' → arr[2] != '\0' → I != '\0' → True → In body increment i → i=3

i=3 → arr[i] != '\0' → arr[3] != '\0' → '\0' != '\0' → False out of loop

Value of i is three that is number of characters in array.

Program 2: Input in char array of size 10 and count number of alphabets and digits in array.

Logic: We shall be using same loop given in above program and in body we shall be applying conditions for alphabets and digit and will be using counters for checking alphabets or digit.

Solution:

```
#include "iostream"
using namespace std;
void main()
{
    char arr[10];
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
cout << "Enter data ";
cin >> arr;
int i = 0;
int count_alpha = 0, count_digit = 0;
while (arr[i] != '\0')
{
    if ((arr[i] >= 'A' && arr[i] <= 'Z') || (arr[i] >= 'a' && arr[i] <= 'z'))
        count_alpha++;
    else if (arr[i] >= '0' && arr[i] <= '9')
        count_digit++;
    i++;
}
cout << "Total Alphabets " << count_alpha << endl;
cout << "Total Digits " << count_digit << endl;
}
```

Program 3: Consider a scenario where you are supposed to store data in employee you have to store name, NIC and contact number. Validate name, NIC and contact number as mentioned below.

- 1) Name should contains alphabets only
- 2) NIC number must be of length 15 or 13. If it is of length 13 than all characters must be digit. If it is of length 15 than their must be 13 digits and at 5 and 13 index their must be –
- 3) Contact Number must start with 03 have all digits and length must be 11.

Logic: We have developed program for checking length, digits and alphabets we shall be using it.

```
#include "iostream"
using namespace std;
void main()
{
    char name[10], nic[16], mblnum[12];
    cout << "Enter name ";
    cin >> name;
    int i = 0, digit = 0;
    while (name[i] != '\0')
    {
        if (name[i] >= 'A' && name[i] <= 'Z')
            i++;
        else if (name[i] >= 'a' && name[i] <= 'z')
            i++;
        else
        {
            cout << "Re-enter name ";
            cin >> name;
            i = 0;
        }
    }
    cout << "Enter NIC ";
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
cin >> nic;
i = 0;
while (true)
{
    if (nic[i] == '\0')
    {
        if (digit == 13 && i == 13)
            break;
        if (digit == 13 && i == 15 && nic[5] == '-' && nic[13] == '-')
            break;
        cout << "Invalid NIC ";
        cin >> nic;
        i = -1;
    }
    else if (nic[i] >= '0' && nic[i] <= '9')
        digit++;
    i++;
}
i = 0;
digit = 0;
cout << "Enter Mobile number ";
cin >> mblnum;
while (true)
{
    if (mblnum[i] == '\0')
    {
        if (digit == 11 && mblnum[0] == '0' && mblnum[1] == '3')
            break;
        cin >> mblnum;
        i = -1;
    }
    else if (mblnum[i] >= '0' && mblnum[i] <= '9')
        digit++;
    i++;
}
}
```

Program 4: Input two arrays and compare data of both arrays if they are equal print arrays are equal otherwise print not equal.

Solution:

```
#include "iostream"
using namespace std;
void main()
{
    char arr1[10], arr2[20];
    cout << "Enter arr 1 ";
    cin >> arr1;
    cout << "Enter arr 2 ";
    cin >> arr2;
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```
int i = 0;
while (arr1[i] != '\0' && arr2[i] != '\0')
{
    if (arr1[i] != arr2[i])
    {
        cout << "Not equal";
        break;
    }
    i++;
}
if (arr1[i] == arr2[i])
{
    cout << "Equal";
}
}
```



Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Lecture # 24 Primitive and Non Primitive Data Types

We shall be discussing primitive data types and non-primitive data types.

Primitive types are also known as built in data types as like int, bool, double etc. What are non-primitive types? Non primitive types are actually called programmer defined data types. These are the data types that a programmer require to make according to his need. When non primitive types are required? Let us discuss a scenario for answer to this question. Suppose that if we want to store data of three students. What will we do for storing data of three students? We shall be making variables for all of them as given below.

Storing record of student

	Student 1	Student 2	Student 3
❖ name	<code>char name[5];</code>	<code>char name1[5];</code>	<code>char name2[5];</code>
❖ age	<code>char address[5];</code>	<code>char address1[5];</code>	<code>char address2[5];</code>
❖ address	<code>int age;</code>	<code>int age1;</code>	<code>int age2;</code>
❖ semester	<code>int smstr;</code>	<code>int smstr1;</code>	<code>int smstr2;</code>
❖ cgpa	<code>float cgpa;</code>	<code>float cgpa1;</code>	<code>float cgpa2;</code>

Why not have a data type that contains name, address, age, smstr, cgpa?

Create Programmer defined data type.

You can see every student has its own variables and we will keep on creating them when we need to store data of more students. If we have a data type whose variable contains name, age, address, semester and cgpa, then it would be very easy for us and we shall be not bothering about number of students. Now How to make programmer defined data type? In C++ with the help of struct we can create programmer defined data type. Syntax is given as follow.

struct name of data type

```
{
```

```
    // name of variables
```

```
    // variables declared here are called data members
```

```
};
```

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

struct is a key word that is used for creating the data type.

Now we create a data type with name student that contains variables as mentioned above.

struct Student

{

```
char name[5];  
char address[5];  
int age;  
int smstr;  
float cgpa;
```



Data Members

};

Now student is a data type. Declaring a variable of student and taking input.

Student s ;

S	
age 	name
smstr 	address
cgpa 	

Figure 1 Variable of student and memory

How do we input in S.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

cin>> S ; is not a valid statement because S holds too many things so we had to specify names of data members. How access data members inside memory S? We can access member inside S using member access operator.

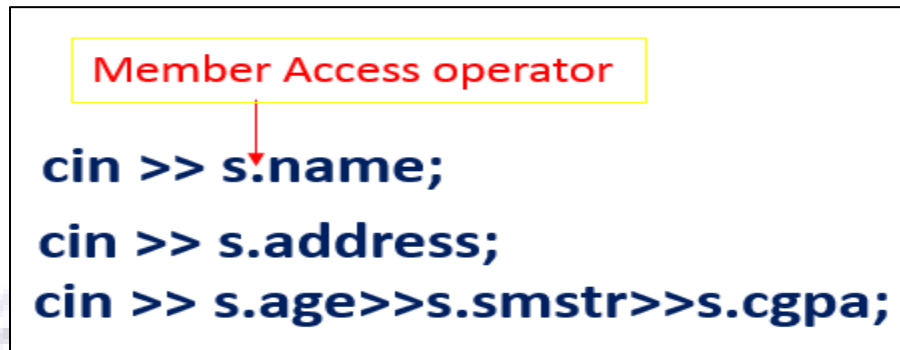


Figure 2 Member Access operator and input in data members

. (period) is referred as member access operator. That is used to access data members.

Program 1: Make a data type student having data members name, address, age, smstr and cgpa make two variable of student take input in both of them

```
#include "iostream"
using namespace std;
struct Student
{
    char name[5], address[5];
    int age, smstr;
    float cgpa;
};
void main()
{
    Student s1, s2;
    cin >> s1.name >> s1.address;
    cin >> s1.age >> s1.smstr >> s1.cgpa;
    cin >> s2.name >> s2.address;
    cin >> s2.age >> s2.smstr >> s2.cgpa;
}
```

s1

age	cgpa	name
21	3.4	A i \0
smstr	address	
3	F 9 \0	

s2

age	cgpa	name
22	3.2	A t t a \0
smstr	address	
5	G 9 \0	

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

If make another will variable Student s3; and assign s1 to s3 note that assignment of structure is same as of primitive data types. But it will copy name of s1 to name of s3 address of s1 to address of s3 cgpa of s1 to cgpa of s3, age and smstr of s1 to age smstr of s3.

Program 2: Make structure student that has data member roll number and cgpa. Make three variables take input and print the one having maximum cgpa.

Program to find Max from 3 integer	Solution to above statement
<pre>#include "iostream" using namespace std; void main() { int num1, num2, num3; cin >> num1; cin >> num2 ; cin >> num3; int max = num1; if (max < num2) max = num2; if (max < num3) max = num3; cout << "Maximum number " << max; }</pre>	<pre>#include "iostream" using namespace std; struct Student { int rolnum; float cgpa; }; void main() { Student s1, s2, s3; cin >> s1.rolnum >> s1.cgpa; cin >> s2.rolnum >> s2.cgpa; cin >> s3.rolnum >> s3.cgpa; Student toper = s1; if (toper.cgpa < s2.cgpa) toper = s2; if (toper.cgpa < s3.cgpa) toper = s3; cout << "Toper " << toper.rolnum; cout << toper.cgpa; }</pre>

Logic is same as for built in data type difference is applying criteria on some data member.

Array of Structure:

Figure given below shows declaration of array and memory. Along with input in array of structures.

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

```

struct Student
{
    int rolnum;
    float cgpa;
};

Static Array
Student s[4];

Dynamic Array
Student* s = new Student[4];
        
```

0	1	2	3
rolnum	rolnum	rolnum	rolnum
cgpa	cgpa	cgpa	cgpa

```

cin >> s[0].rolnum >> s[0].cgpa;
cin >> s[1].rolnum >> s[1].cgpa;
cin >> s[2].rolnum >> s[2].cgpa;
cin >> s[3].rolnum >> s[3].cgpa;
        
```

```

for (int i = 0; i < 4; i++)
    cin >> s[i].rolnum >> s[i].cgpa;
        
```

Figure 2 Array of structure

Above figure explains how to declare array of structures and access input using loop and without loop.

Program 3: Declare a four size array of student with data members roll number and cgpa find and print data of toper.

Max from array of integers	Solution to above Statement
<pre> #include "iostream" using namespace std; void main() { int arr[4]; for (int i = 0; i < 4; i++) cin >> arr[i]; int max = arr[0]; for (int i = 1; i < 4; i++) { if (max < arr[i]) max = arr[i]; } cout << "Max " << max << endl; } </pre>	<pre> #include "iostream" using namespace std; struct Student { int rolnum; float cgpa; }; void main() { Student s[4]; for (int i = 0; i < 4; i++) cin >> s[i].rolnum >> s[i].cgpa; Student max = s[0]; for (int i = 1; i < 4; i++) { if (max.cgpa < s[i].cgpa) max = s[i]; } cout << "Max " << max.rolnum << max.cgpa; } </pre>

Northern University, Nowshera

Programming Fundamentals (ECS-121)

Muhammad Athar

email id: athar@northern.edu.pk,

Whatsapp# 0333-5077664

Program 4: You are supposed to store result of Student. Make structure result having data members roll number, subject, marks. Input roll number of a student, search and display its data.

Solution:

```
#include "iostream"
using namespace std;
struct Result
{
    int rnum, marks;
    char subj[5];
};
void main()
{
    Result* r = new Result[3];
    for (int i = 0; i < 3; i++)
        cin >> r[i].rnum >> r[i].marks >> r[i].subj;
    int search;
    cout << "Enter roll number";
    cin >> search;
    for (int i = 0; i < 3; i++)
    {
        if (r[i].rnum == search)
            cout << r[i].rnum << r[i].marks << r[i].subj;
    }
}
```