

String lab Exercise

1. Write a C++ program which displays (your name) a string entered by user.

```
#include <iostream>
#include<string>
using namespace std;
int main()
{
    string s;
    cout<<"Enter Your name"<<endl;
    cin>>s;
    cout<<"your name="<<s;
}
```

Output

```
Enter Your name
Abdi
your name=Abdi
-----
```

Or by using char data type

```
#include <iostream>
#include <string>
using namespace std;

int main ()
{
    char str[100];
    k: cout <<"Enter a string ";
    cin>>str;
    cout << "You Entered : " << str << endl;
    cout << "Enter another string:";
    cin>>str;
    cout << "You Entered : " << str << endl;
    goto k;

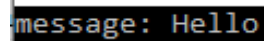
    return 0;
}
```

```
Enter a string
Mark
You Entered : Mark
Enter another string:
Zukerberg
You Entered : Zukerberg
Enter a string
Elemo
You Entered : Elemo
Enter another string:
Qiltu
You Entered : Qiltu
Enter a string
```

2. Initialize and display a string

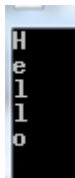
```
#include<iostream>
using namespace std;
int main ()
{
char G[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
cout << "message: "<<G;
return 0;
}
```

Output

A terminal window showing the output of the first program. The text "message: Hello" is displayed on a black background with white text.

★ Initialize and display a string using for loop statement

```
#include <iostream.h>
#include <conio.h>
int main()
{
    char str[ ] = {'H','e','l','l','o','\0'};
    int i;
    for( i=0; i<6; i++)
    {
        cout << str[i]<<endl;
    }
    getch();
    return 0;
}
```

A terminal window showing the output of the second program. The characters 'H', 'e', 'l', 'l', and 'o' are displayed on separate lines on a black background with white text.

Write c++ program which illustrates string concatenate,copy and length

```
#include <iostream.h>
#include <conio.h>
int main ()
{
char str1[20] = "Fundamental";
char str2[20] = "programming";
char str3[20];
// copying str1 into str3
cout << "After copying the string ="<< strcpy( str3, str1)<< endl;
// concatenates str1 and str2
cout << "After Concatenation string will be:" << strcat( str1, str2)<< endl;
// total length of str1 after concatenation
cout << "Length of string after concatenation : " << strlen(str1) << endl;
//string compare
cout<<"String 1 ="<<str1<<endl;
getch();
return 0;
}
```

```
After copying the string =Fundamental
After Concatenation string will be:Fundamentalprogramming
Length of string after concatenation : 22
String 1 =Fundamentalprogramming
```

❖ General syntax for string compare

strcmp(str1,str2);

- ★ if s1 and s2 are the same → returns 0
- ★ if s1<s2 → returns negative
- ★ if s1>s2 → returns positive

If the two strings are **identical**, it returns a **0**. If not, then it returns the difference between the ASCII values of the first non-matching pairs of characters.

Pointer lab Exercise

Exercise

1. Assign address of a variable to pointer variable then display their values and address.
2. Apply arithmetic operation of division operator to pointer
3. Display the address of pointer variable before and after incrementing.

Note:-size of integer data type is 4 bytes.

Solution

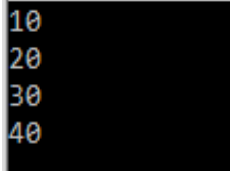
```
#include<iostream>
using namespace std;
int main()
{
    int a = 10;
    int *ptr1;
    ptr1=&a;
    cout<<"value of a="<<a<<endl;
    cout<<"Address of a="<<&a<<endl;
    cout<<"Value of pointer ="<<*ptr1<<endl;
    *ptr1=*ptr1/2;
    cout<<"Value of pointer after division operator"<<*ptr1<<endl;
    cout<<"Address of pointer(ptr1) before incrementation : "<<ptr1<<endl;
    ptr1++;
    cout<<"Address of pointer (ptr1) after incrementation : ";
    cout<<ptr1<<endl;
}
```

Output

```
value of a=10
Address of a=0x6ffe04
Value of pointer =10
Value of pointer after division operator=5
Address of pointer(ptr1) before incrementation :0x6ffe04
Address of pointer (ptr1) after incrementation : 0x6ffe08
```

4. Initialize and display array elements using pointers.

```
#include<iostream>
using namespace std;
int main()
{
    int a[] = {10, 20, 30, 40, 50};
    int *p;
    p = &a[0]; // we can assign as p=a, but impossible assigning as p=a[0] or p=&a or p=&a[]
    cout<<*p<<endl;
    p++;
    cout<<*p<<endl;
    p++;
    cout<<*p<<endl;
    p++;
    cout<<*p<<endl;
    return 0;
}
```



```
10
20
30
40
```

Output

Note:-P++ indicates location of next position of array.

Optional

★ Initialize and display array elements using pointers by for loop statement.

```
#include<iostream>
using namespace std;
int main()
{
    int *ptr1;
    int a[] = {10, 20, 30, 40, 50, 60};
    ptr1 = &a[0];
    for(int i=0; i<5; i++)
    {
        cout<<*(ptr1+i)<<endl;
    }
}
```

Output

```
10
20
30
40
50
60
```

5. A program to illustrate the use of arithmetic in pointers.

***++ptr, ++*ptr and *ptr++**

```
#include<iostream>
using namespace std;
int main()
{
    int a[4]={10,20,30,40};
    int *ptr;
    ptr=a;
    cout<<"Content of Array A: "<<endl;
    cout<<*ptr++<<" "<<endl;
    cout<<*ptr<<" "<<endl;
    cout<<++*ptr<<" "<<endl;
    cout<<*++ptr<<" "<<endl;
    cout<<*ptr--<<" "<<endl;
    cout<<*ptr<<" "<<endl;
    cout<<*--ptr<<" "<<endl;
    return 0;
}
```

Output

```
Content of Array A:
10
20
21
30
30
21
10
```

Exercise

6. Write a C++ program to accept N integer values that stored in an array using a pointer from keyword .Then print the elements of the array on the screen.

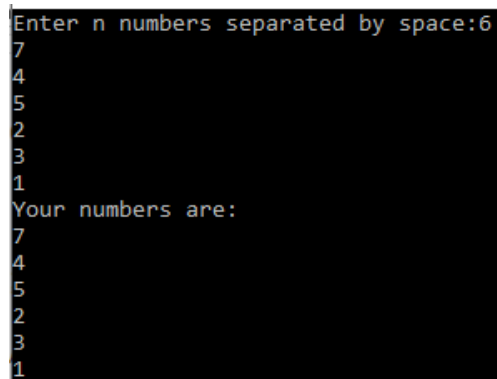
Solution:

```
#include<iostream>
using namespace std;

int main()
{
    int arr[50], n;
    int *p;
    p=arr;      //or p=&arr[0]
    cout<<"Enter n numbers separated by space:";
    cin>>n;
    for(int i=0; i<n; i++)
        cin>>*(p+i); //cin>>arr[i] or //cin>>*p>>*(p+1)>>*(p+2)>>*(p+3)>>*(p+4);
    cout<<"Your numbers are:\n";
    for(int i=0; i<n; i++)
        cout<<*(p+i)<<endl;    // or we can use cout<<arr[i]<<endl;
    return 0;
}
```

Note:- $a[2] \longleftrightarrow *(a + 2)$

Output



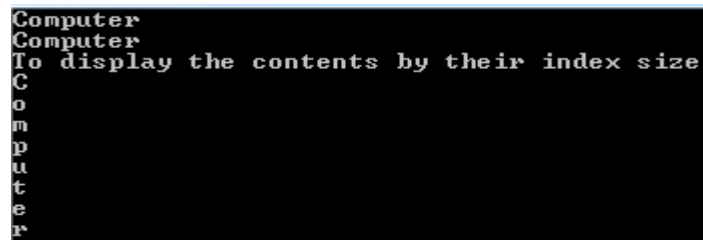
```
Enter n numbers separated by space:6
7
4
5
2
3
1
Your numbers are:
7
4
5
2
3
1
```

Exercise

Initialize and display character array using pointers

Exercise 2

```
#include<iostream.h>
#include<conio.h>
void main()
{
char a[]="Computer";
char *cp;
cp=a;
cout<<a<<endl;
cout<<cp<<endl;    //is equivalent to an array a.
cout<<"To display the contents by their index size \n";
cout<<*cp<<endl; // cout<<a[0] or cout<<*(cp+0);
cout<<*(cp+1)<<endl;    //a[1] or cout<<a[1];
cout<<*(cp+2)<<endl;    //a[2]
cout<<*(cp+3)<<endl;    //a[3]
cout<<*(cp+4)<<endl;    //a[4]
cout<<*(cp+5)<<endl;    //a[5]
cout<<*(cp+6)<<endl;    //a[6]
cout<<*(cp+7)<<endl;    //a[7]
getch();
}
```



```
Computer
Computer
To display the contents by their index size
C
o
m
p
u
t
e
r
```

Output

Pointer to pointer or double pointer

Initialize values to a variable which illustrates double pointer or pointer to pointer.

```
#include<iostream.h>
#include<conio.h>
void main()
{
    int data=50;
    int *ptr;
    ptr=&data;
    cout<<"Value in *ptr is:"<<*ptr<<endl;    //single pointer
    cout<<"****DEMO FOR POINTER TO POINTER *****\n";
    int **pptr;
    pptr=&ptr;
    cout<<"Value in **pptr is : "<<**pptr;    //double pointer
    getch();
}
```

Output

```
Value in *ptr is:50
****DEMO FOR POINTER TO POINTER ****
Value in **pptr is :50
```

Exercise

Declare a pointer to pointer variable and display the contents of these pointers.

```
#include<iostream>
using namespace std;
int main()
{
    int a;
    int *ptr1, **ptr2;
    cout<< "Enter any number : ";
    cin>>a;
    ptr1 = &a;
    ptr2 = &ptr1;
    cout<<"The entered number is : "<<a;
    cout<<endl<<"The value of first pointer is : "<<*ptr1;
    cout<<endl<<"The value of second pointer is : "<<**ptr2;
    return 0;
}
```

Output

```
Enter any number : 50
The entered number is : 50
The value of first pointer is : 50
The value of second pointer is : 50
-----
```