Home Work No.15 Pointers-1

////////////////////////// Lecture 1 on Pointers 29-11-2022

Intro to pointers:

1. **Pointer is a variable, which stores address of another variable.**
2. **Pointer is Power programming tool and Pointer is a two faced sword**
    a. **Variable value**
    b. **Variable address (pointer own value)**
3. **Size of pointer**
    - **Size of pointer depends upon System architecture.**
    - **Pointer size does not depend upon data type.**
    - **Question: Why we use type with pointers?**
4. **Relationship between pointer and data.**
5. **Same type pointer variable points to same type data.**
6. **Pointer has close relationship with arrays and strings. Using pointers, we can access arrays and string flexibly and convenient.**
7. **If passed in functions Pointers simulates pass by reference.**
8. **Pointer are used for Dynamic memory allocation. It means runtime memory allocation.**
9. **How to use pointers.**

Pointer deceleration:

```
Data_Type  * PointerName;
```

Assigning address to a pointer:

```
PointerName = &VariableName;
```

Pointer Assignment:

Two ways of assignment:

1. **No address assignment or NULL/0**
2. **Point a variable/Array Address**

10. **Accessing values of pointer (Accessing Address of other variable)**
11. **Indirection/Dereference (Accessing data)**

Run this program understand it carefully

```
#include <iostream>

using namespace std;

int main()
{

	int a = 10;
	char c = 'a';


	int * aptr = NULL;//No address in pointer
	char *cptr = 0;//No address in pointer


	aptr = &a;//Integer varible address stores in pointer
	cptr = &c;


	cout << endl;

	cout << "\nAdress of variable \"a\" OR Value of pointer \"aptr\" is  "<<aptr <<"
::::  "<<&a <<endl;

     cout << "\nDerefernece or Indirection through pointer or variable \"a\" is  " <<
*aptr << "  :::::   " <<a<<endl;//Indirection/Dereference opertor

	cout << "\nAdress of variable \"c\" OR Value of pointer \"cptr\" is " << cptr << "
" << &c << endl;
```

```
	cout << "\nDErefernece or Indirection through pointer or variable \"c\"  is  " <<
*cptr << "  " << c << endl;//Indirection/Dereference opertor



	cout << "\nSize of Char pointer" << sizeof(cptr);//Size of Character Pointer
	cout << "\nSize of int pointer" << sizeof(aptr);//Size of integer pointer



}
```