University of Engineering and Technology,
Peshawar, Pakistan

CSE 102: Computer Programming

Lecture 06

Pointers

By;
Dr. Muhammad Athar Javed Sethi

Definition

- Variables that hold the address values
- Every byte has an address
- Addresses are numbers
- Every variable & function starts at a particular address

Pointer Declaration

- char* cptr;
- int* iptr;
- float* fptr;

The Address-Operator (&)

- Used to find the address of variable

```
int var1 = 11;
cout<<&var1<<endl;
```

Output: 0x8f4ffff4

Pointer Variables

- A variable that hold an address value is called a pointer variable

```
int var1 = 11;
int* ptr;
ptr = &var1;
cout<<ptr<<endl;
```

Output: 0x8f4fdfc4

Pointer variable…

- We can also display the value stored at a particular address with the use of pointers

```
int var1 = 11;
int* ptr;
ptr = &var1;
cout<<*ptr<<endl;
```

Output: 11

Pointer to void

- A pointer to void is general purpose pointer that can point to any data type

```
int intvar;
float flovar;
void* ptrvoid;
ptrvoid = &intvar;
ptrvoid = &flovar;
```

Pointer and Arrays

- There is a close association between pointers and arrays
- Array elements can be accessed using array notation as well as pointer notation

```
int intarray[5] = { 31, 54, 77, 52, 93 }
```

Array notation:

```
for(int j = 0; j < 5; j++)
    cout<<intarray[j]<<endl;
```

Pointer and Arrays…

Pointer notation:

```
for( int j = 0; j < 5; j++ )
  cout<< *(intarray + j) << endl;
```

Pointers and Functions

- We can pass arguments to a functions by;
  1)- By value
  2)- By reference or By pointer
- Arguments are passed by value if they are not modified in the original program

Pointers and Functions…

- Arguments are passed by reference or by pointer if we want to modify the value in the original program
- Passing arguments by pointer is similar to passing arguments by reference

Example…

- By Pointer:

```
void square(int*);
void main(){
int var = 10;
cout<<var<<endl;
square(&var);
cout<<var<<endl;}
```

Example…

```
void square(int* v){
        v  = v * v;}
```

Output:

10

100

Pointers to Pointers

- A pointer can also hold the address of a pointer variable i.e. A pointer to a pointer holds the address of another address

```
int var =  10;
int* ptr1 = &var;
int** ptr2 = &ptr1;
cout<<*ptr1<<endl<<*ptr2;
```

Output: 10 ← value at address stored in ptr1

0x8f4fdfc4 ← value contained in ptr2