POINTER BASICS

A pointer variable is a variable that holds the address of another variable.

Syntax

```
int*  p;            OR          int  *p;       //An integer pointer
char*  ch;           OR          char  *ch; // A character pointer
```

Usage

```
int* p;
int i;
p=&i;   // & is called the address of operator
```

After the code above:

p contains the address of i
p “points to” i
i = the variable pointed to by p
i = *p
*p is the content of the memory location pointed to by p
* is called the *dereferencing operator*, which is a unary operator (The symbol * is also used for the binary operation of multiplication)

Member Access Operator: Suppose Student is a class with a member variable gpa of type double. Then we can create a pointer to the class Student:

```
Student* sp;
Student s1;
sp = & s1;
```

The statement   (*sp).gpa = 3.7; is equivalent to   s->gpa=3.7;

Dynamic Variables: Pointers are most useful with dynamic variables. Dynamic variables have the following two features different from ordinary variables:

- They are created during the execution of a program, not in compile time
- They have no identifiers.

Syntax

```
 int* p;
 p = new int;

 int* q;
 q = new int[10];
 q[0] = 5;
 q[5] = 25;
*( q+5) ;   //what is this equal to?
```

Delete operator: When a dynamic variable is no longer needed, delete operator is used to free the memory a dynamic variable occupies.

```
int* p;                                  int* q;
p = new int;                            q = new int[10];
………..                                    …………………

delete p;                               delete [ ] q;
```