CSCI 2270 Lecture Notes

1/23/19

Pointers

* a variable type for storing a memory address
* int \*p;
  + the int specifies what type of data it points to
  + the \* indicates the variable is a pointer type
* int \*p = &x;
  + stores the address of x into the pointer p
* Where to put the \* when doing variable declarations?
  + int\* p;
  + int \*p;
    - preferred method. Can declare regular and pointer int type variables in the same line of code this way, specifying a pointer with the \* in front of the appropriate variable down the line
  + int \* p;
* How do we get target value?
  + int \*p = &x;
    - cout << \*p << endl;
      * This is called the dereference operator (\*). Dereferencing the pointer means “go to this address and give me what is in their”
    - cout << p << endl;
      * The memory address p is storing, in this case the address of x
* Pass by Pointer
  + void myPassByPointer(int \*p); – function declaration
    - this function is expecting a pointer type variable
    - myPassByPointer(myPtr);
  + like with pass by reference, can also do
    - myPassByPointer(&x);
  + a function can also return a pointer type
    - int \*foo(){

int x = 3;

int \*p = &x;

return p;

}

* + - this will return the address p points to, not the value stored there