CHAPTER 7
POINTERS

CMPS 148

Pointers

- Pointers are **variables** that hold the memory address of **another** variable
- Declare variables using the * modifier
  - Also called a "reference" variable
  - `int x;  // regular variable`
  - `int * y; //pointer to an (undefined) integer`

Working with Pointers

- To make a pointer "point" to something, you need to get its address

```
int x; // regular variable
int * y = &x;
```

- To change the value of x:

```
x = 5; // or
*y = 5; // "dereferences" y
```

Dynamic Allocation

- You can dynamically allocate variables using pointers

```
int * y = new int;
```

  - y points to a new, un-named integer

```
int *z = y;
```

  - z now points to the same un-named integer

- Variables allocated with new do not belong to a particular function - they are in the **heap**

Deleting memory

- Regular variables are destroyed when their **scope** ends
- Dynamic variables do not - we must manually delete them

  delete y;

  - now what does z point to?
    - `*z = 5; //` illegal, our program might crash!
    - dangling pointer

Memory Leaks

```
int *y = new int;
int z;
y = &z;
```

- Uh oh! How do we delete our dynamic variable?
- Pointer are very useful, but also very dangerous!

Dynamic Arrays

- We can create new arrays using pointers:

```
int x = 50
… user enters a value for x
int myArray[x];
// Error:  Size must be known at compile time

int * myArray = new int [x];
…
delete [] a;
```

Arrays as Pointers

- Note the syntax in the previous slide…
  - C++ array names **are** pointers
  - `int list[10];`
  - `list[2] = 5`
  - `*(list+2) = 5`

Lab #5

- Write a program that allows the user to type **any** amount of input numbers (-1 to stop):
  - Print out the sorted result
- Start by creating a **dynamic array** of size 5.
- Whenever the user enters too many numbers, grow the size of the array by 2
  - For example, when the 6th number is entered:
    - Create a new dynamic array of size 10
    - Transfer the existing 5 numbers
    - **Delete** the old array, change the pointer.
    - Add the 6th number