CS246 Assignment 0

Nissan Pow

May 11, 2006

Part 1: C++ Basics

1. In the declaration statement *int* a = 3*, the *** indicates that *a* is a pointer to an int. In the assignment *b = *a*, the *** is the dereference operator, and returns the object pointed to by the pointer (ie the value of the pointer).

In the declaration statement *int& a = 3*, the *&* indicates that *a* is a reference to another object of type int. In the assignment statement *b = &a*, the *&* returns the address of *a*.

2. A "statically declared variable" is created on the stack.

3. Code to dynamically declare a 2-D array and initialize it to form a 10x10 identity matrix.

```
#include <iostream>
using namespace std;

#define DEBUG

int main() {
  int **array;
  int size = 10;
  array = new int*[size];
  for (int i=0; i<size; ++i) {
    array[i] = new int[size];
  }
  for (int i=0; i<size; ++i) {
    for (int j=0; j<size; ++j) {
      array[i][j] = (i==j ? 1 : 0);
    }
  }
#ifdef DEBUG
  for (int i=0; i<size; ++i) {
    for (int j=0; j<size; ++j) {
```

```
      cout << array[i][j];
    }
    cout << endl;
  }
#endif
  // cleanup
  for (int i=0; i<size; ++i) {
    delete[] array[i];
  }
  delete[] array;
  return 0;
}
```

4. The code will output: *"All Conditions Fail"*.

if *((int)a == b)* is comparing the address of *new int(100)* with 100, and the address is very unlikely to be 100 so this condition will fail.

if *((int)&b == c)* is comparing the address of *b* with 100, so it will fail with the same reason as above.

if *((int)a != b && (int)&b != c)* will be true, since both of the above are false.

Part 2: Reproduce a UML Diagram