

ECE2036: Week 6 – Arrays and Pointers

Reading: Chapter 7-8

Most of our discussion has been on the board. In this handout we have two program examples to trace through. We will include the idea of dynamically allocating elements, using pointers to pass-by-reference, reference declarations, and C-style pointers.

```
//-----  
//----- Basic Trace Through Example  
//-----  
#include <iostream>  
using namespace std;  
  
int main()  
{  
  
    int a[8] = {1,2,3,4,5,6,7,8};  
    int *aptr = NULL;  
  
    aptr = &a[0];  
    a[2] = a[2]+1;  
    a[3] = a[3]+a[4];  
    aptr++;  
    (*aptr)++;  
    (*(++aptr))--;  
    cout << a[1] << a[2] << a[3] << a[4] << *aptr << endl;  
}  
//-----  
//----- Example of Using Pointers to Pass-by-Reference  
//-----  
#include <iostream>  
using namespace std;  
  
void cloudingFunctionsPurpose1(int *);  
void cloudingFunctionsPurpose2(int *);  
  
int main()  
{  
    int *firstPtr;  
    int firstValue = 42;  
  
    int *secondPtr;  
    int secondValue = 72;  
  
    firstPtr = &firstValue;  
    secondPtr = &secondValue;  
  
    firstPtr = &(*secondPtr);  
  
    (*firstPtr)++;  
  
    cloudingFunctionsPurpose1( &firstValue);  
    cloudingFunctionsPurpose2( secondPtr);  
  
    cout << *firstPtr << *secondPtr << "[ ]" << --(*(&firstValue)) << endl;  
    cout << (*(&firstValue))++ << endl;  
  
    return 0;  
}  
void cloudingFunctionsPurpose1 ( int * cloudingVariable)  
{ *cloudingVariable /= 2; }  
  
void cloudingFunctionsPurpose2 ( int * cloudingVariable)  
{ cloudingVariable--; }
```