Lab 6: Changing to C++
Pointers

Reading: Chapter 7 - Pointers
Chapter 15 – C++ as a better C

Turn in:

1) **Lab 6 Questions Handout, on paper**
2) **lab6.cpp – submit on Canvas**
3) **makefile – submit on Canvas**

-

C++ compiler

```
g++ myprogram.cpp
```

Compiler options such as –o and –c are the same as for gcc.

C++ Library pg 552-553	Similar C Library
#include <iostream>	#include <stdio.h>
#include <iomanip>	
#include <cmath>	#include <math.h>
#include <cstdlib>	#include <stdlib.h>
#include <ctime>	#include <time.h>
#include <cctype>	#include <ctype.h>

In C++	**In C**
`int main( )`	`int main (void)`
Declare variables	same
Assignment statements	same
If statements	same
while loop	same

The loop control variable may be declared inside the loop "initialization" section.
`for (int i = 1; i < 10; i++)`
	`int i;`
	`for (i = 1; i < 10; i++)`

pointers	same
functions	same
`cout << x << " some text " << endl;`	`printf("%d some text \n", x);`
`cin >> y;`	`scanf("%d", &y);`

1. **Create a file called lab6.cpp. Copy the following code into your program, then fill in lines of code to implement the tasks described in the comments.**

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


// write the prototype for a function called swap that accepts two
// integer pointers as parameters.  The function will swap the contents
// of the two addresses.   The function will not return anything.
// Hint:  this function appears on page 293


// write the prototype for a function called maximum.  The function will
// accept two parameters:  a pointer to a float, and an integer.
// The integer represents the size of an array and the float pointer
// is the address of the first array element.
// The function will find and return the maximum value in the array.

// write the prototype for
      void printNperline ( int array[ ] , int size, int n )
// (This function performs the same task as the function in Lab 5.  However, you
// must change the output statements to C++.



// start the main function


// define an array of integers called values with 20 elements


// using a for loop, assign the even, positive integers (starting with 2)
// to the array elements


// declare an integer named x and initialize it to 1


// declare an integer named y and initialize it to 2


// call the swap function using addresses of  values[x] and values[y]


// print a message "After the first swap:\n".
// using the printNperline function, print the contents of the array values,
// 10 elements per line

// print a blank line


// define a pointer named aPtr that points to an object of type int
// assign the starting address of the array values to aPtr

// define a pointer named vPtr that points to an object of type int
// assign vPtr the address of values[1]
// define a pointer named wPtr that points to an object of type int
// assign wPtr the address of values[2]


// call the swap2 function using parameters vPtr and wPtr


// print a message "After the second swap:\n".
// using the printNperline, print the contents of the array values,
// 10 elements per line

// print a blank line


// print a message "Using Pointer/Offset Notation:\n"
// using a for loop and pointer/offset notation, print the contents
// of the array values, all on one line


// print a message "Using Pointer subscripting:\n"
// using a for loop and pointer subscripting, print the contents
// of the array values, all on one line


// print the pointer aPtr (with an informative message)


// print the pointer aPtr + 3 (with an informative message)


// print the value stored at aPtr + 3 (with an informative message)


// declare an array called list with 10 elements of type float
// using a for loop, input numbers from the user and store them
// in the array list


// call the maximum function with the array list and 10.  Store the return
// value in a variable called max.
// print max (with an informative message)


// end the main function


// write the definition of the swap function (see the description
// above)

// write the definition of the maximum function (see the description
// above)

// write the definition of the printNperline function (see the description above)
```

2. **Create a makefile to compile your program and produce an executable called lab6. Be sure to include the "all" and "clean" targets.**

3. **Compile, test, and debug your program as needed.**

4. **Obtain a copy of the Lab 6 Questions handout (given in class on Feb 24).**

5. **Submit 2 files on Canvas**

   1) **.cpp file with the C++ source code**
   2) **makefile**

6. **Submit the completed Lab 6 Questions handout on paper by the end of lab on March 5, 2:00 pm.**

   **** you may turn it in early at SH 157 or slide it under the door if no one is in the office**