

**These must be completed and shown to your lab TA either by the end of this lab, or by the start of your next lab.**

The following is an introduction to C++ through a variety of simple activities. You should also read the C++ Primer in your textbook and practice as much as possible.

1. Compile and run the following two programs that demonstrate input and output in C++.

Program 1 (terminal I/O):

```
#include <iostream>
using namespace std;
float circle_area(float radius); // declare function before main()
int main(void) {
    float circle_radius;
    cout << "Enter radius:" << endl;
    cin >> circle_radius;
    cout << circle_area(circle_radius) << endl;
    return 0;
}
float circle_area(float radius) {
    return 3.14159 * radius * radius; // = pi * r^2
}
```

Program 2 (file I/O):

First create a file called `infile.txt` containing a few lines of text.

```
#include <iostream>
#include <string>
#include <fstream>
using namespace std;
int main(void) {
    ifstream in("infile.txt"); // Open for reading
    ofstream out("outfile.txt"); // Open for writing
    string s;
    while(getline(in, s)) // Discards newline char
        out << s << "\n"; // ... add it to the new file
    cout << "End of program\n";
    return 0;
}
```

2. Declare an array with 10 elements. Write a function `fill_array` to fill the elements with the numbers 1-10. Modify the function so that it accepts two integers. The first integer represents the first element, the second integer represents the increment between each element. For example `fill_array(4,2)` will fill the array with the numbers 4, 6, 8, 10, 12, 14, 16, 18, 20, 22. Print the contents of the array to the screen once it has been filled.
3. (from Wikipedia) The Tower of Hanoi or Towers of Hanoi is a mathematical game or puzzle. It consists of three pegs (called *A*, *B* and *C*), and a number *n* of disks of different sizes which can slide onto any peg. The puzzle starts with the disks neatly stacked in order of size on peg *A*, the smallest at the top, thus making a conical shape.

The objective of the puzzle is to move the entire stack to peg *C*, obeying the following rules:

- Only one disk may be moved at a time.
- Each move consists of taking the upper disk from one of the pegs and sliding it onto another peg, on top of the other disks that may already be present on that peg.
- No disk may be placed on top of a smaller disk.

Your task is to write code to solve the Towers of Hanoi problem for  $n$  disks. Hint: Use recursion!

Recursively move  $n - 1$  disks from  $A$  to  $B$ . This leaves disk  $n$  alone on peg  $A$ .

Move disk  $n$  from  $A$  to  $C$ .

Recursively move  $n - 1$  disks from  $B$  to  $C$  so they sit on disk  $n$ .

Your program should take a small integer  $n$  as input from the command line using code like this:

```
int main(int argc, char* argv[]) {
    if( argc != 2 ) {
        cerr << "Usage: " << argv[0] << " n" << endl;
        return -1;
    }
    int n = atoi(argv[1]);
    moveDisks(n, "A", "B", "C");
    return 0;
}
```

It should produce output to solve the problem with  $n$  disks, The output of my `hanoi 3` is:

```
Move disk from peg A to peg C
Move disk from peg A to peg B
Move disk from peg C to peg B
Move disk from peg A to peg C
Move disk from peg B to peg A
Move disk from peg B to peg C
Move disk from peg A to peg C
```

4. Write a program that simulates a guessing game. It should randomly generate a number (look up `rand` and `srand` using Google for help on random numbers) and ask the user to input a guess. The game should keep running until the user gets the number correct, or otherwise indicates that they wish to end the game.
5. Be sure to show your work to your TA before you leave, or at the start of the next lab, or you will not receive credit for the lab!