# CS 111 Final Programming Test

This programming assignment is designed to assess your C++ programming proficiency. The test is closed note but you are encouraged to look up and memorize the answers beforehand.

Unless otherwise specified, you are not allowed to use built-in "magic" functions that solve the problem using a single or very few lines of code. When in doubt, ask Adam.

## Requirements

For this lab, you must implement the following functions in C++. Note that each function is graded independently of each other. If you are not comfortable writing function, you may place each task in its own file for reduced credit.

**TIP**: it is always better to make an attempt than to submit something empty. I am a person too; I want you to succeed! It's much easier to give sympathy points when there's at least evidence that you tried.

### Function #1: find\_largest

Implement a function called ***int find\_largest(vector<int>\* some\_list)*** that finds the largest number present in some\_list.

Example Usage:

|  |
| --- |
| vector<int> data{1, -3, 5, 2, 9, 1, 0};  int largest = find\_largest(&my\_list);  cout << largest << endl; //output is 9 |

### Function #2: write\_list

Implement a function called ***void write\_list(vector<int>\* some\_list, string file\_name)*** that writes the contents of the list to the file name present in some\_list (one item per line in the file).

Example Usage:

|  |
| --- |
| vector<int> data{1, -3, 5, 2, 9, 1, 0};  string file\_name = "output.txt";  write\_list(&my\_list, file\_name); #outputs contents of my\_list to file |

### Function #3: read\_list

Implement a function called ***vector<int> read\_list(string file\_name)*** that reads the contents present in file\_name into a vector. Your function should return this list back to the caller. The file will contain one entry per line.

Example Usage:

|  |
| --- |
| string file\_name = "function3.txt";  vector<int> items = read\_list(file\_name);  //items now contains a bunch of numbers |

### Function #4: convert\_time

Implement a function called ***string calculate\_time(int hour, int minute, int second)*** that converts the supplied hours, minutes, and seconds from a 24 hour clock to a 12 hour clock (AM / PM) and returns the appropriate string. Assume valid input ranges (0-24 for hours, 0-60 for minutes, 0-60 for seconds). Examples:

* If your function is supplied the values 6, 12, 30, your function should return "06:12:30AM" as a string.
* If your function is supplied the values 13, 0, 1, your function should return "01:00:01PM".

Example Usage:

|  |
| --- |
| string time = calculate\_time(6, 12, 30);  cout << time << endl; //will print "06:12:30AM"  time = calculate\_time(13, 0, 1);  cout << time << endl; //will print "01:00:01PM" |

### Function #5: to\_upper

Implement a function called ***string to\_upper(string text)*** that converts the supplied string into all upper case. Ignore all other characters. For example, "Hello!" should be converted into "HELLO!"

Example Usage:

|  |
| --- |
| string upper = to\_upper("Hello!");  cout << upper << endl; |

## Grading

Each function will be evaluated on the following 10-point scale (50 points overall):

* [1] A function is used
* [1] Functions return the correct value to the caller (no prints!)
* [8] The function meets the stated requirements. This is broken down as follows:
  + [8] – The function works exactly as specified by the requirements
  + [6-7] – The function is nearly correct, but has one minor flaw
  + [4-5] – The function is mostly correct or has a few flaws in its implementation
  + [3] – The function is missing significant portions of its implementation
  + [1-2] – An attempt was made