CS 104 (Fall 2013) — Assignment 1
Due: 09/05/2013, 11:59am

BitBucket directory name for this homework (case sensitive): HW1

(1) Carefully read the course web site, `http://www-bcf.usc.edu/~dkempe/CS104/index.html`. Then answer the following questions:

(a) Which of the following are acceptable behaviors in solving homeworks/projects?

[ ] Looking up information relevant to the course online.
[ ] Looking up or asking for sample solutions online.
[ ] Talking to my classmates about the problems.
[ ] Copying code from my classmates, and then editing it significantly.
[ ] Asking the course staff for help.

(b) Which of the following are recommended ways of writing code?

[ ] gedit
[ ] emacs
[ ] Eclipse
[ ] vim
[ ] Microsoft Visual Studio
[ ] notepad

(c) Which C++ compiler do you have to use?

[ ] I have to use G++.
[ ] I can use any compiler I want, so long as it works on my machine.
[ ] I can use any compiler I want, but the submission has to work with G++ in the end.

(2) Set up your BitBucket account, set all permissions, give access to your teaching team, and get ready coding with version control. In particular, you will be submitting this (and future) homeworks through Git/BitBucket.

(3) Carefully review recursion and dynamic memory management from your CSCI 101 notes and textbook. You may also find Chapters 2 and 5 from the textbook, and the C++ interlude 2, helpful.

(4) For each of the following two programs, rewrite the central loop using recursion instead of a loop. Use the timing techniques (see the handout of useful stuff) to measure how long the processing of the loop takes with both methods (`for` loop and recursion). Do not include the time to read the input. From this, infer roughly how much overhead you get from recursion compared to loops (as a percentage). Make sure that your arrays are large enough, and you run enough iterations, to make your measurements accurate. Submit both your code and the measurements you have taken.

(a)
```
/* various declarations here */

int main (void)
{
  int *a;
  int n;  // number of elements in the array a
  int sum; // we'll use this for adding numbers
  /* here goes code to read n numbers into a, probably from a file
     Make sure to allocate space to a
  */

  // the next part is the loop you are supposed to replace by recursion.

  sum = 0;
  for (int i = 0; i < n; i ++)
      sum += a[i];

  // This is the end of the loop.

  cout << sum << endl;
  return 0;
}
```

(b)

```
/* various declarations here */

int main (void)
{
  int *a;
  int n;  // number of elements in the array a
  int m; // we'll use this for the maximum

  /* here goes code to read n numbers into a, probably from a file
     Make sure to allocate space to a
  */

  // the next part is the loop you are supposed to replace by recursion.

  m = a[0];
  for (int i = 1; i < n; i ++)
      if (a[i] > m) m = a[i];

  // This is the end of the loop.

  cout << m << endl;
  return 0;
}
```