CS142 **Practice for Test 2** Spring 2025

Books, notes, calculators, computers and phones are *not* permitted.

If a question asks for C++ code, don't worry about remembering every little detail of C++ syntax. Minor details will not affect your grade.

1. (16%) Answer each of the following questions briefly but precisely.

   (a) What is the main advantage of integration testing (over testing all the components together right away)? [Section 4.3]

   (b) What are two operations that are supported by random-access iterators but not by bidirectional iterators? [Section 5.3]

2. (20%) Create a function `concatenate(v1, v2, v3)` that takes as arguments three vectors of integers `v1`, `v2` and `v3` and makes `v3` contain a copy of all the elements of `v1` followed by a copy of all the elements of `v2`. The original contents of `v3` is deleted. Implement the function as efficiently as possible. [Exercise 4.2.10]

3. (24%) Implement the generic algorithm `max_element(start, stop)`. Recall that this algorithm returns an iterator that points to the maximum element in the range `[start,stop)`. If the range is empty, `stop` is returned. Elements are compared by using the < operator. The arguments `start` and `stop` are bidirectional iterators. [Exercise 5.6.4(b)]

4. (20%) Implement the generic algorithm

   ```
   copy_if(start, stop, dest, condition)
   ```

   Recall that this algorithm copies the elements in the range `[start,stop)` to a range of positions that begins at `dest`. The only elements that are copied are those that satisfy the unary predicate `condition`. If *n* is the number of elements that are copied, an iterator that points *n* positions to the right of `dest` is returned. Copies forward, from `start` to `stop`. The arguments `start`, `stop` and `dest` are bidirectional iterators. [Exercise 5.8.4(d)]

5. (20%) Create a function `make_double_spaced(ls)` that takes as argument a list of strings and inserts a new empty string after every string in the list. [Exercise 6.4.5]