SE2852 Exam 1 Feedback

Dr. Yoder, Spring 2014, MSOE

Problem 4, Page 3

① Returning T[] ref will auto-cast to Object[] ref. Could declare temp array variable to be of type Object[] ref instead. See also ③.

② ArrayLists consider "null" elements to be just as valid as non-null elements.

③ Cast from T[] ref to Object[] ref can be done implicitly. Similarly, cast from T to Object can be done implicitly. The cast from T[] to Object[] is only safe because of type erasure – after compiling, T[] is really only a reference to an Object[] anyway. Generally, if B extends A, the cast from B[] to A[] will compile and run, but can result in unexpected Run-time errors.

④ -10 "This method must allocate a new array, even if this list is backed by an array." This means you must create a new array, and copy the data into it.

⑤ -3 The array that should be copied is an instance variable. It is not passed in as an argument.

⑥ -1 Use the .length property to get the length of an array. Arrays (e.g. Object[]) do not have a .size() or .length() method. Use `a[i]` rather than `a.get(i)` to access an element in an array.

⑦ -2 Set the size of an array using the syntax `Object[] array = new Object[size];` You cannot resize an array once it is created.

⑧ -1 By default, a newly-created array ref (e.g. `Object[] a;`) is not initialized or is null. Initialize it by instantiating a new array and assigning the reference to `a`, e.g. `a = new Object[size];`

⑨ -1 The method should return a reference to the new `Object[]` as specified in the API.

Problem 8a, Page 6

① -1 worst-case: O(n) is a measure of the worst-case running time for an algorithm, not an average-case or sum-of-all cases running time.

② -1 Simplify the O(n) expression