String and IntList

Most students did well on both of these problems. To look for memory leaks I augmented the supplied tests for each with some simple long-running versions of the supplied tests. Six students failed at least one leak checker for String. The same was true for IntList but it wasn't the same six students. The full set of tests used for grading can be found in $FILES/a7.

answers.txt

Question (1), regarding references and operator overloading was a tough one. Only a few people nailed it, but any sort of reasonable discussion was worth four points. If largely correct, it was worth all five points.

A few people said that references are definitely need to implement assignment, because (by convention) it modifies the left hand operand. That's incorrect. Remember that x = y is equivalent to x.operator=(y) which invokes the **member function** operator=(...), and member functions have implicit access to the object's data (unless they are declared const.)

Another somewhat common response was to simply argue that references allow the operator implementations to change the operand objects(s). That's true, but so would passing a pointer.

For question (2), regarding a warning on mixing new/delete with malloc()/free(), I was hoping that by now most everybody would have picked on the theme of C++ generally favoring performance over any sort of protection, and would have blasted the proposal out of the water. A number of students fired away, but lots of others liked it. However, any thoughtful argument earned all five points.

Miscellaneous

There were 14 extra.txt submissions that had a clear estimate of hours spent on the whole assignment and that didn't cite factors that could add significant noise to the data. The average was 5.6 hours; the median was 4 hours.