

Program #1 Grading Criteria - 100 pts total

Code Points (25 pts)

Reads in multiple test cases - 1 pt

Uses structs as presented in the assignment - 2 pts

Each of 4 function prototypes followed - 8 pts total, 2 pts each, all or nothing for each function

Each function returns correct TYPE of information - 4 pts total, 1 pt for each function

Strings are read into a fixed size array, then a dynamically allocated array is created of the right size to store the string and the string is copied in - 5 pts total (give 3 pts if strcpy is used but the malloc is always for 21 chars)

Attempts to free memory - 5 pts

Feel free to award partial credit in each category based on “how well” the program satisfies the criteria presented.

Style Points (10 pts)

Header comment w/name, program, date – 2 pts

Appropriate variable names – 2 pts

Appropriate white space and indenting – 2 pt

Internal comments in code – 4 pts

Process Document (15 pts)

Planning Phase - 5 pts

Assistance - 0 pts

Debugging Phase - 5 pts

Testing Phase - 5 pts

Be fairly lenient here, but make sure this corresponds to their work. But the quality is way worse than what you expect, please take off some points.

Execution Points (50 pts)

10 pts for doing stdin, stdout

There are five test files: coursesummary_sample, coursesummary_bigcase1, coursesummary_bigcase2, coursesummary_bigcase3, coursesummary_bigcase4.

each is worth 8 points. You can award partial credit on any file. Just eyeball how many things look right and proportionally score out of 8. Alternatively, if you debug and fix it, you may take off points for the actual error depending on how big you think it is.

DO NOT SPEND MORE THAN 5 MINUTES LOOKING FOR AN ERROR. IF YOU CAN'T FIND IT, JUST GRADE BASED ON THE ACTUAL TEST CASES.

Extra Credit

Add 5 pts for using leak detector. Add 5 more points if they have no memory leaks when run with leak detector.