CSSE 132 – Introduction to Computer Systems
Rose-Hulman Institute of Technology
Computer Science and Software Engineering Department

Homework 4

When writing code, make sure you document it well with comments. This is an individual assignment.

1. On your Pi, update your SVN repository to find a "homework4" directory containing some files including `problems.c`. Complete the unfinished functions in `problems.c`. When you're finished, **commit your solutions in `problems.c` to SVN.** Suggestions:

    - The comments in `problems.c` tell you what needs to be completed.
    - Delete the `TODO` comments when you complete the relevant exercises.
    - Use `make test` to compile your homework and the tests:

      ```
      pi@student-pi:~/1516c-csse132-student/homework4$ make test
      gcc -c -o unity/unity.o -Iunity unity/unity.c
      gcc -g -o test test.c problems.c unity/unity.o
      pi@student-pi:~/1516c-csse132-student/homework4$
      ```

    - Once built, run the `test` binary to test your code (the example below shows an unfinished assignment):

      ```
      pi@student-pi:~/1516c-csse132-student/homework4$ ./test
      HW4 test.c:39:test_length:FAIL: Expected 0 Was -1
      HW4 test.c:48:test_lengthNoArrays:FAIL: Expected 0 Was -1
      HW4 test.c:57:test_bitDropper:FAIL: Expected 8 Was -1
       ... testing "e" ...
      HW4 test.c:75:test_upcaseVowels:FAIL: Expected 1 Was -1
      HW4 test.c:105:test_skipper:FAIL: Expected Non-NULL

      -----------------------
      5 Tests 5 Failures 0 Ignored
      FAIL
      pi@student-pi:~/1516c-csse132-student/homework4$
      ```

    - The tests provided in tests.c are not exhaustive. Consider adding more tests to test.c (though it is not required).
    - The `switch` statement in C may come in handy.