NCWIT Engage CS Edu Implementation Paper

Use programming to analyze real human DNA files

ELIZABETH BOESE, University of Colorado - Boulder

This paper provides guidance to instructors who would like to implement this project. It provides information on the practicalities of making interdisciplinary connections. It also points out areas where students may struggle and how an instructor may respond. The paper concludes by discussing three additional items: the use of pair programming with this assignment, how to effectively implement the "problem solving" portion of the project, and the opportunity the project presents to discuss testing.

1. OVERVIEW

This assignment provides CS1 students a chance to write a program in the world of genetics. Specifically this assignment looks at skin type, type-2 diabetes, exercise and diet based on real human DNA files. It includes references to a website with a diagram showing how the genotypes for exercise and diet interrelate and students need to develop code to implement the diagram. Learning objectives include: command-line arguments, data structure (python dictionary), if-else, loops, file input, writing user-defined functions.

2. TEACHING DNA

This assignment is a great opportunity to bring in a guest lecturer from any field that works with DNA (biology, microbiology, computational biology, bioinformatics, computer science, etc.) Grad students work great and appreciate the opportunity to present in front of a class. This also enables a networking opportunity for students who find this assignment interesting and wish to pursue either an interdisciplinary major or major/minor in one of the sciences and computer science. Video options for teaching about DNA:
https://www.youtube.com/watch?v=uXdzuz5Q-hs
https://www.youtube.com/watch?v=BmDG_fKUTR8

3. STUDENT STRUGGLES

There are a couple of “gotcha” traps that students fall in to during this assignment. Use this for testing to ensure they do it right, or discuss in lecture, or modify the assignment as desired.

3.1 Ignoring comment lines

The assignment states, “If the line starts with a ‘#’ then that line is a comment so skip over it.” However, many students ignore this directive and instead count how many comment lines are at the top of the file and skip only those. I give them a data file where I comment out one of lines in the data file, as well as a small test file that only has 2 comments at the top of the program.

3.2 To dictionary, or not to dictionary

Many students skip storing the data in the dictionary (or other data structure you use for your programming language).

3.3 Removing the newline

When reading in lines from python, we end up with the newline character at the end of the string. Other languages may or many not have this issue.

3.4 The difficulty of the extra credit

The diagram is difficult to implement and has a lot of nested if statements. To simplify, focus on exercise first in its own if statement then work on diet. Also make use of the variables (see hint in assignment) as that helps tremendously!

3.5 Learning how to use small test files

Students will jump in and start using the massive real human DNA files, which takes a while every time they go to test their program. This is a great opportunity to teach them to build a small test file themselves to work with while in development, then use the real data files to test their assignment when they think they are done.

4. PAIR PROGRAMMING

This assignment works great for pair programming as they are trying to understand both DNA and the new concepts of user-defined functions. Also, if they attempt a divide-and-conquer technique for accomplishing the assignment, both should still end up meeting the learning objectives.

5. PROBLEM-SOLVING

The portion on exercise and diet is a complicated one, and helps to use pen and paper to figure out how to translate the diagram to code. One thing to point out is when the same rsid is used in multiple sections of the diagram. This is great for discussing why it is easier to make use of variables before entering in to the if-structure. Encourage or help students walk through the paths, and to walk through their code and ensure there are no missing branches.

6. TESTING

This assignment would also be an excellent time to discuss testing. There are many variations through the simple ones (diabetes and skin type) and a more complex set of test cases for diet and exercise. For example, you could add an assignment/lab exercise/in class exercise to have them create multiple data files that tests for all scenarios, which is another great option for pair programming or in-class group work.

ACKNOWLEDGMENTS

Martin Muggli, for introducing me to his research and the 23andme website which inspired this assignment, and being a guest speaker for my class to teach the basics of DNA.

FURTHER READING

23andMe website to get your own DNA data: https://www.23andme.com/
23andMe file info: https://www.snpedia.com/index.php/23andMe
Report details of a DNA data set: http://www.snpedia.com/index.php/Promethease
Public 23andMe data files:

- [various – be sure data type says 23andme] (sometimes takes a while to load) https://my.pgp-hms.org/public_genetic_data
- [various – more samples – use .txt files] http://stanford.edu/class/gene210/files/data/Genomes/
- Mikolaj Habryn http://sites.google.com/a/rcpt.to/dichro/Home/genotyping/genome_Mikolaj_Habryn_20080522154706.zip?attredirects=0