UDACITY

Verified Certificate Evaluation

STUDENT Jay Shah

COURSE Introduction to Computer Science

Excellent work so far, Jay! I really like reading through your code. It makes use of good helper procedures, and it has numerous comments that guide you through each procedure. It's thought-out, Pythonic, and functions fantastically.

I would only like to see an update on the style of the code. Docstrings are used for every procedure in your code, and this is great. Make sure they follow the Python style guidelines, though; you can find those here: http://legacy.python.org/dev/peps/pep-0257/#multi-line-docstrings. For most of your docstrings, which are multiple lines long, there should be """ (double quotes, rather than single quotes) surrounding them. Also, subsequent lines should be on new lines. See the link for details.

I couldn't help but pick up on a few other little things that caught my eye. In the `path_to_friend` procedure, you have what's known as 'dead code': code that has been commented out. It's a good idea to remove this, since it clutters up code. Finally, since indentation is so important in Python, it's important to follow it strictly. I noticed a couple of lines with extra tabs in the `add_connection` and `get_connections`. If you look through your code one more time, I'm sure you can make all the indentation consistent across the entire program.

Again, truly good work! I'm really nitpicking here when it comes to your final project, since it is essentially done. I'm looking forward to seeing those beautiful docstrings!

Code Functionality **Meets Specifications**

- Code passes the autograder when it is submitted
- Code completes the desired task and uses a data structure which allows for efficient execution of the code even with large inputs.

Use of Control Statements **Meets Specifications**

- Selection of control statements is rarely inappropriate

- Looping control statements are appropriately used to avoid repetitive code
- Branching control statements are used correctly in a way that is not difficult to understand
- Appropriate Python syntax is often used with control statements, making them more intuitive

Use of Procedures **Exceeds Specifications**

- Repetitive blocks of code are contained within their own procedures
- Recursion is used correctly in the problem that asks for it
- Choice of arguments and return values is aligned with the purpose of the procedure

Make-Your-Own-Procedure (MYOP) **Meets Specifications**

- MYOP is well planned, designed, and produces a useful result
- Comments clearly explain MYOP usage and functionality

Code Readability **Does Not Meet Specifications**

- Most variables and helper procedures are given meaningful and useful names
- Appropriate variables are used instead of hard-coded values, clarifying code logic
- **Docstrings are present, but are not correctly formatted.**
- **Code has commented-out lines ('dead code')**

PROJECT EVALUATION

Project Does Not Meet Specifications