Week 6: Individual Progress Report

GitHub URL: https://github.com/dathaotrinh/WaterCarrier

- **No** - your product is effective to delivery the knowledge in computer science to the product users.
    - My product applies a popular technique in computer science to calculate the maximum water a container can carry. The users can use my product to improve their math skills but my product is not purely used to deliver knowledge in computer science.
- User friendly/appealing in terms of the following criteria
    - **YES** - The landing page is attractive
        - I have not designed a landing page for my product yet. However, this will be a goal in the next milestone.
    - **YES** - Users are be able to understand and play the puzzle game quickly.
        - I'm planning to add the instructions to play the game at the landing page so that the users can understand the concept quickly.
    - **YES** - Users can just jump in and start playing (trying out) the game immediately without the registration process.
        - Users do not have to have an account to play the game, but only authenticated users are able to record their game records.
- Your product should have the following functions.
    - **YES** - Users can register with a username and a password.
        - Yes, this is a must for this project.
    - **YES** - The performance of registered users are updated after each trial and can be displayed upon requests.
        - The game ranking will be shown on the leaderboard.
        - This is planned to be implemented in the next milestone (milestone 4)
    - **YES** - Users can ask for hints and/or solutions.
        - This is planned to be implemented in the next milestone (milestone 4)
    - **YES** - Administration account
        - Have all the functionality like the regular registered users.
        - Have additional privilege likes user account removals or password-reset.
        - This is planned to be implemented in the next milestone (milestone 4)

- **YES** - Do you have a brute-force method as the comparison basis for the puzzle solver.
    - Calculate the area created by all wall pairs.
- **YES** - Do you have a better algorithm than brute-force.
    - Start with two pointers, one at the beginning of the array and the other at the end.
    - Calculate the area formed by the current two pointers.
    - Move the pointer with the shorter height inward, as the area is limited by the shorter height. This is because increasing the distance between the two lines can only increase the area if the height of the shorter line increases.
- Explain if the puzzle is targeted at a single user or multiplayer, competitive or non-competitive.
    - The puzzle is targeted at a single user. Each user will play his/her game and solve the puzzle independently.
    - Depending on how quick the user answers the question and the difficulty level of each game, a score will be given.
- Explain how to deploy your product.
    - So far I've found some tutorials to deploy my Flask project to PythonAnywhere's free server (YouTube tutorials attached). However, PythonAnywhere mentioned on their website that using SQLite for the backend may cause a little delay and slow the system. I need to do more research on this and may consider switching to MySQL.
        - https://www.youtube.com/watch?v=qSuMP3jifbk
        - https://www.youtube.com/watch?v=q8I1CPKjGrU
        - https://help.pythonanywhere.com/pages/KindsOfDatabases/