Test Balancing Cases

- Exactly 2 containers on one side of ship
    - Success, *computes in <1 second*
- Unbalanced ship with more containers on left side
    - Success, *computes in <1 second*
- Unbalanced ship with more containers on right side
    - Success, *computes in <1 second*
- Exactly 6 containers on one side of ship
    - Success, *computes in <1 second*
- Containers stacked to ship height in one column
    - Success, *computes in <1 second*
- Containers all along bottom edge of ship
    - Success, *computes in <1 second*
- Exactly 4 containers on one side of ship
    - Success, *computes in <1 second*
- Ship with 80 containers needs balancing
    - Success/Fail (Depending on distribution of containers), *computes in ~11 minutes when successful*
- Empty ship that needs balancing
    - Success, *computes in <1 second*
- Ship with single container needs balancing
    - Success, *computes in <1 second*
- Ship that is already balanced
    - Success, *computes in <1 second*
- Ship that requires sifting
    - Success, *computes in <1 second*
- Ship that requires stacking above 9 high
    - Success, *computes in <1 second*

Test Load/Unload Cases

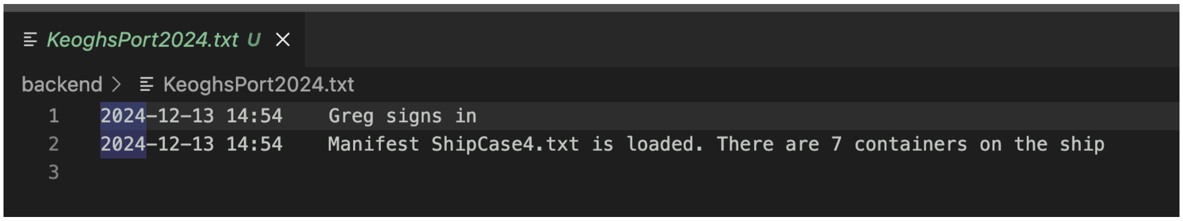
- Exactly 2 containers next to each other on one side of ship and unloading one
    - Success, *computes in <1 second*
- Exactly 2 containers next to each other on one side of ship and unloading both
    - Success, *computes in <1 second*
- Exactly 4 containers spread out in ship and loading one container
    - Success, *computes in <1 second*
- Exactly 4 containers spread out in ship and loading 4 containers
    - Success, *computes in <1 second*
- Exactly 6 containers with some stacked on one side of ship and loading two containers and unloading one container simultaneously

    - Success, *computes in <1 second*
- Containers stacked to ship height in one column with one load and one unload
    - Success, *computes in <1 second*
- 3 columns of containers stacked to max capacity with unloading a container from the left most column
    - Success, *computes in <1 second*
- 3 columns of containers stacked to max capacity with unloading a container from the rightmost column
    - Success, *computes in <1 second*
- 3 columns of containers stacked to max capacity with unloading a container from the middle column
    - Fail, *returns runtime error*
- Containers all along bottom edge of ship with nothing on top with 2 loads and 2 unloads simultaneously
    - Success, *computes in <1 second*
- Containers all along bottom edge of ship with another row on top with 2 loads and one unload at topmost row and one unload at bottom most row
    - Success, *computes in <1 second*
- Ship filled with maximum capacity and attempting to load
    - Unhandled
- Ship filled with maximum capacity and unloading bottom most container
    - Fail, *returns runtime error*
- Ship filled with maximum capacity and unloading top most container
    - Success, *computes in <1 second*
- Operator sends one load but forgets to select other container to be loaded (2 separate loads)
    - Success, *computes in <1 second*
- Operator sends on unload but forgets to select other container to be unloaded (2 separate unloads)
    - Fail, *incorrectly identifies container name*

Test API Calls

- Logging in
    - Success
        - "OPTIONS /login HTTP/1.1" 200 -
        - "POST /login HTTP/1.1" 200 -
- Uploading the manifest through load/unload
    - Success
        - “POST /fileUploadLoad HTTP/1.1" 200 -
        - "GET /fileUploadLoad HTTP/1.1" 200 -

        - "GET /fileUploadLoad HTTP/1.1" 200 -
        - "GET /get_fileName HTTP/1.1" 200 -
        - "GET /get_fileName HTTP/1.1" 200 -
- Uploading the manifest through balance
    - Success
        - "GET /fileUploadBalance HTTP/1.1" 200 -
        - "GET /checkbalance HTTP/1.1" 200 -
        - "GET /fileUploadBalance HTTP/1.1" 200 -
        - "GET /checkbalance HTTP/1.1" 200 -
        - "GET /get_fileName HTTP/1.1" 200 -
        - "GET /get_fileName HTTP/1.1" 200
- Writing a comment to log
    - Success
        - "OPTIONS /comment HTTP/1.1" 200 -
        - {'comment': 'dent'}
        - "POST /comment HTTP/1.1" 200 -
- Generates a log file / appends to log file keoghsport2024
    - Success



```
2024-12-13 14:54    Greg signs in
2024-12-13 14:54    Manifest ShipCase4.txt is loaded. There are 7 containers on the ship
```

        - 127.0.0.1 - - [13/Dec/2024 14:54:51] "OPTIONS /login HTTP/1.1" 200 -
        - 127.0.0.1 - - [13/Dec/2024 14:54:51] "POST /login HTTP/1.1" 200 -
        - 127.0.0.1 - - [13/Dec/2024 14:54:55] "POST /fileUploadLoad HTTP/1.1" 200 -
        - 127.0.0.1 - - [13/Dec/2024 14:54:55] "GET /fileUploadLoad HTTP/1.1" 200 -
        - "GET /fileUploadLoad HTTP/1.1" 200 -
        - 127.0.0.1 - - [13/Dec/2024 14:54:55] "GET /get_fileName HTTP/1.1" 200 -
        - 127.0.0.1 - - [13/Dec/2024 14:54:55] "GET /get_fileName HTTP/1.1" 200 -
- Generates a manifest for load/unload
    - Success
        - "POST /load HTTP/1.1" 200 -
        - “GET /manifest HTTP/1.1" 304 -
- Generates a manifest for balance

- Success
    - "GET /manifest HTTP/1.1" 200 -
    - "GET /get_fileName HTTP/1.1" 200 -