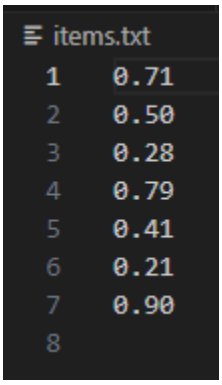
Bin Packing Problem Multi-Algorithm Manual

Ethan Shelstad

Running the Algorithm

- Open the Linux terminal or Linux terminal emulator of your choice.
- Navigate to the folder where the GitHub repository is accessible
- Ensure that there is a .txt file named “items.txt”
    - The file should remain in the following format:



    - However, changes to the file can be made as long as
        - All numbers remain as doubles
        - There is at least one double
        - Each line has precisely one double
        - No double exceeds 1.0
    - The following changes can be made to
        - Doubles can be removed
        - Doubles can be added
        - Doubles can be edited
- Once the list of Items is set Run the command “make main”.

```
shepinel@DESKTOP-AA6KLAN:/mnt/c/Users/e_she/OneDrive/Documents/GitHub/project-3-eksinel$ make main
g++ -std=c++11 -Wall -Wextra -c main.cpp -o main.o
g++ -std=c++11 -Wall -Wextra -c sort.cpp -o sort.o
g++ -std=c++11 -Wall -Wextra -c fitCheck.cpp -o fitCheck.o
g++ -std=c++11 -Wall -Wextra main.o sort.o fitCheck.o -o main
shepinel@DESKTOP-AA6KLAN:/mnt/c/Users/e_she/OneDrive/Documents/GitHub/project-3-eksinel$
```

- If the file for any reason does not properly make, ensure all files are compiled in the same repository and no changes have been made to the core files.
- To run the algorithm, Run the command “./main”
- Results should appear similar to the below
    - If any changes are made to the items.txt file, the results will appear differently.
- To run again with different values, simply make the changes to the “items.txt” file (following the rules listed above), save the “items.txt” with the new changes, and run the “./main” command again.
    - It is not necessary to run the “make main” command again.

```
shepinel@DESKTOP-AA6KLAN:/mnt/c/Users/e_she/OneDrive/Documents/GitHub/project-3-eksinel$ ./main items.txt
Total items: 7
Policy              | Total Bins Used
--------------------|-----------------
Optimal Solution    | 4
Online - First Fit  | 4
Online - Next Fit   | 5
Online - Best Fit   | 4
Offline - First Fit | 4
Offline - Best Fit  | 4

Optimal:
Bin 0: 0.79, 0.21
Bin 1: 0.28, 0.71
Bin 2: 0.41, 0.50
Bin 3: 0.90

Online - First Fit:
Bin 0: 0.71, 0.28
Bin 1: 0.50, 0.41
Bin 2: 0.79, 0.21
Bin 3: 0.90

Online - Next Fit:
Bin 0: 0.71
Bin 1: 0.50, 0.28
Bin 2: 0.79
Bin 3: 0.41, 0.21
Bin 4: 0.90

Online - Best Fit:
Bin 0: 0.71, 0.28
Bin 1: 0.50, 0.41
Bin 2: 0.79, 0.21
Bin 3: 0.90

Offline - First Fit:
Bin 0: 0.90
Bin 1: 0.79, 0.21
Bin 2: 0.71, 0.28
Bin 3: 0.50, 0.41

Offline - Best Fit:
Bin 0: 0.90
Bin 1: 0.79, 0.21
Bin 2: 0.71, 0.28
Bin 3: 0.50, 0.41
```

How to read results

- The results are broken into *three* categories
- The Total Items:
    - Displayed at the top of the results, the total items will display the number of doubles listed in the item sizes
- The algorithm used and the total bins used
    - A table displays the algorithm used on the left and the number of bins that were used in that algorithm on the right.
    - The brute force algorithm is displayed as the optimal solution
        - Compare the results of other algorithms to the optimal solution to compare accuracy. The closer the number of bins used, the more optimal the solution.
- Visualization of the Bins and their contents Organized by algorithm
    - Each algorithm is titled and separated by blank lines
    - Each line displays an individual bin, followed by the item sizes that compose that bin