Part 3: Comparing the Cow Transport Algorithms

Bookmark this page

Problem Set due Nov 4, 2022 07:30 +08 Completed

POLL: ALGORITHM INTUITION

(ungraded) Before doing the task in this part, answer the following question to see your intuition for how the greedy and brute force algorithm run. In terms of time, which algorithm do you expect will run faster?

RESULTS

Greedy Algorithm	**93%**
Brute Force Algorithm	**7%**

Submit

Results gathered from 454 respondents.

Part 3: Compare the Algorithms

Implement `compare_cow_transport_algorithms`. Load the cow data in *ps1_cow_data.txt*, and then run your greedy and brute force cow transport algorithms on the data to find the minimum number of trips found by each algorithm and how long each method takes. Use the default weight limits of 10 for both algorithms. Make sure you've tested both your greedy and brute force algorithms before you implement this!

Hints:

- You can measure the time a block of code takes to execute using the `time.time()` function as follows. This prints the duration in seconds, as a float. For a very small fraction of a second, it will print 0.0.

```
start = time.time()
## code to be timed
end = time.time()
print(end - start)
```

- Using the given default weight limits of 10 and the given cow data, both algorithms should not take more than a few seconds to run.

Part 3-1

2.0/2.0 points (graded)

Now that you have run your benchmarks, which algorithm runs faster?

- (●) The Greedy Transport Algorithm
- ( ) The Brute Force Transport Algorithm
- ( ) They take the same amount of time

✔

Submit You have used 1 of 1 attempt

Part 3-2

2.0/2.0 points (graded)

Consider the properties of the GREEDY algorithm. Will it return the optimal solution?

- ( ) Yes, all the time
- ( ) No, never

Calculator

It could, but it depends on the data, not always.

Submit You have used 1 of 1 attempt

Part 3-3

2.0/2.0 points (graded)

Consider the properties of the BRUTE FORCE algorithm. Will it return the optimal solution?

- Yes, all the time
- No, never
- It could, but it depends on the data, not always.

Submit You have used 1 of 1 attempt

Calculator