## Subgroup B

### B.1 — [S01B1] Apply surge/penalty rules (conditionals)

Scenario (e-commerce):

Context:

Pricing in the e-commerce app uses a base per-km rate and time-based surge after business peaks. Product wants a deterministic calculator for receipts and audits.

Your Task:

Implement a fare function: fare = km \* base\_per\_km \* surgeMultiplier, where surge applies strictly after 18:00 local time.

Data & Edge Cases:

Input is a list of rides with `time` as HH:MM (24h) and `km` as float. Edge case: exactly at 18:00 should be treated as non-surge for 18:00:00; after 18:00 (e.g., 18:01) surges.

AI Assistance Expectation:

Prompt AI to outline parsing HH:MM, applying conditionals, and rounding to 2 decimals; then implement and write a quick test.

Constraints & Notes:

No external libraries; round each fare to 2 decimals; do not mutate input.

Sample Input

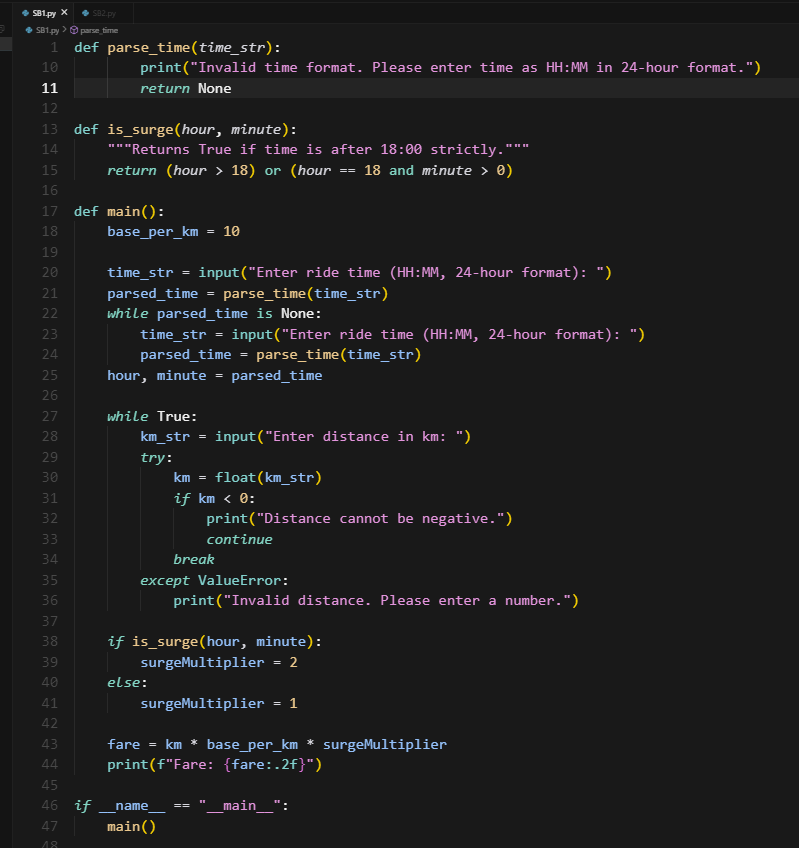
[{'time': '08:00', 'km': 3.0}, {'time': '18:30', 'km': 5.0}]

Sample Output

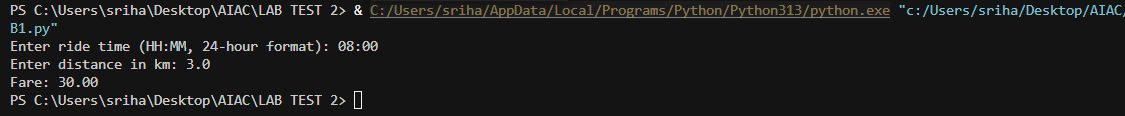
[30.0, 60.0]

Acceptance Criteria: Correct surge threshold and rounding

Code:



Output:



### B.2 — [S01B2] Debug rolling mean (off-by-one)

Scenario (e-commerce):

Context:

A team in e-commerce noticed off-by-one bugs in a rolling KPI computation (moving averages) that undercount windows.

Your Task:

Use AI to identify the bug and fix the window iteration so all valid windows are included.

Data & Edge Cases:

For xs=[1, 2, 3, 4] and w=2, number of windows should be len(xs)-w+1.

AI Assistance Expectation:

Ask AI to add a failing test first, propose the minimal fix, and verify with the sample.

Constraints & Notes:

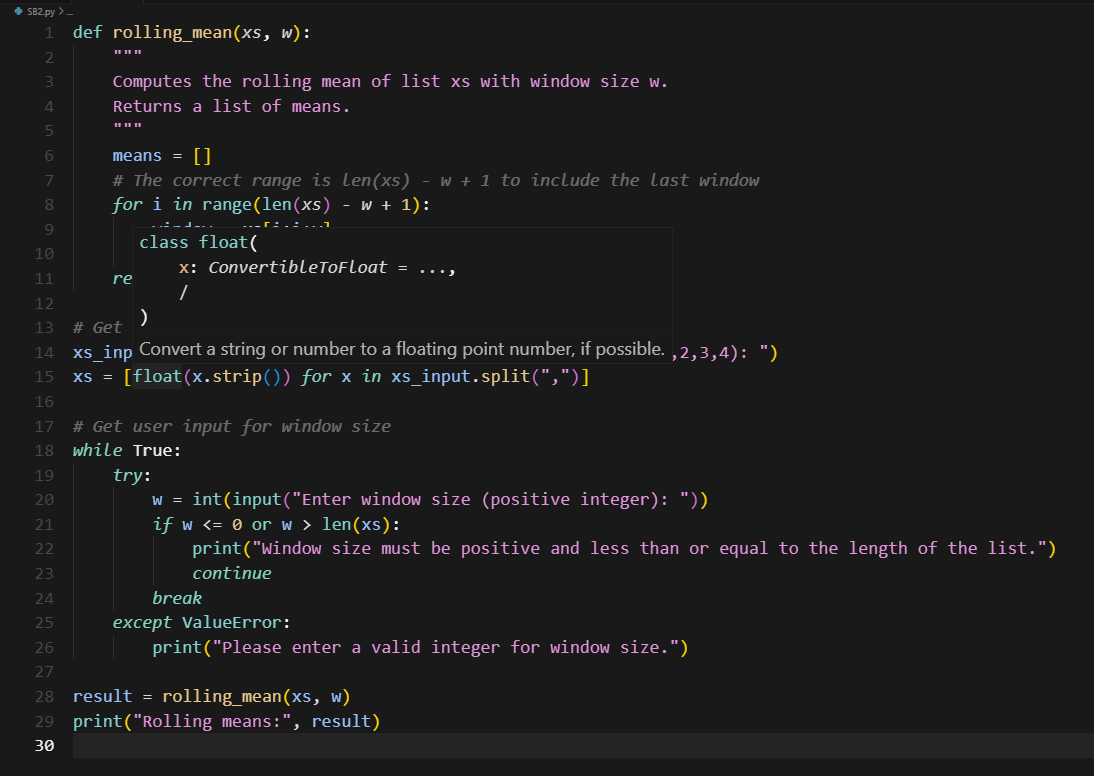
Guard invalid w (<=0 or >len(xs)); preserve O(n\*w) simple solution.

Sample Input

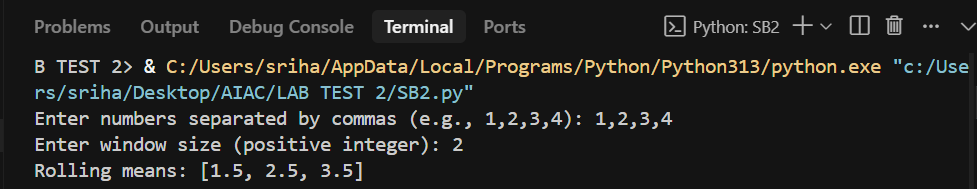
xs=[1, 2, 3, 4], w=2  
Buggy code:  
  
def rolling\_mean(xs, w):  
 sums = []  
 for i in range(len(xs)-w):  
 window = xs[i:i+w]  
 sums.append(sum(window)/w)  
 return sums  
Sample Output

[1.5, 2.5, 3.5]

Acceptance Criteria: All valid windows included; passes tests; no index errors



Output:



------------------------------------------------------------