As Judy suggested in the cubicle conversation, if there are multiple high score solutions with the same cost, this code favors the first solution it finds. But why is that? What aspect of the code makes that happen? What if you wanted to favor the last solution you found instead; how would you do that?

Answer:
Because this code is always comparing the current low cost by using less than,
so once a low cost is established,
it takes an even lower cost (not an equal cost) to pick a new winner.
You could favor the last solution it sees by changing the less than comparison to a less than or equal to, <=, comparison.

Answer: Because this code is always comparing the current low cost by using less than, so once a low cost is established, it takes an even lower cost (not an equal cost) to pick a new winner. You could favor the last solution it sees by changing the less than comparison to a less than or equal to, <=, comparison.