

Show all posts by recent activity

? [Lab 5: Exercise C- Wrong Answer](#) 5
[Hi, My Brute-force algorithm...](#)

💬 [Lab 5 exercise C Distance for Bruteforce](#) 3
[Hello the distance for the br...](#)

? [Lab 6: Strange Results](#) 6

? [a headache question in lab5 exercise C](#) 4
[Hi, for the exercise C of the L...](#)

💬 [Lab 5, C](#) 3
[Dear all, I cannot make exer...](#)

✓ [How to I read the adjacency graph?](#) 6
[Hi, I am stuck at exercise C o...](#)

💬 [Lab 5 \(c\)](#) 3

? [lab 5 Exercise C](#) 21
[Where the instructions say:...](#)

? [lab 5 Exercise C](#) 18
[In the description of the gre...](#)

Lab 5 exercise C Distance for Bruteforce

discussion posted 7 months ago by [Ateesh](#)

Hello the distance for the brute force is coming as 94 not 81, request your help, attached is the code:

```
infinity = 10000000

def greedy_complete_graph(graph, initial_vertex):
    current_distance = 0

    # while we still have a vertex that is not explored
    # get next target using choose_target function
    # update player position
    # update distance value
    # remove next target from vertices list
    #
    vertices=list(range(len(graph)))
    vertices.remove(initial_vertex)
    current_initial_vertex = initial_vertex
    while (vertices!=[]):
        next_target=choose_target(graph[current_initial_vertex],vertices)
        current_distance+=graph[current_initial_vertex][next_target]
        current_initial_vertex=next_target

        vertices.remove(next_target)
    return(current_distance)

best_distance = infinity

def bruteforce_complete_graph_auxiliary_function(remaining_vertices, current_vertex,
current_distance, graph):

    global best_distance
    # if there are no remaining vertices:
    # if the current distance is shorter than the best distance:
    # update the best distance to match the current distance
    # else:
    # for each vertex in the remaining vertices:
    # perform a copy of the remaining vertices, let us call them "new_vertices"
    # remove the vertex from new_vertices
    # call the function bruteforce_complete_graph_auxiliary_function recursively
    #
    if len(remaining_vertices)==0:
        if current_distance < best_distance:
            best_distance= current_distance
    else:
        for i in remaining_vertices:
            new_vertices = remaining_vertices.copy()
            new_vertices.remove(i)
            current_distance = current_distance+ graph[current_vertex][i]
            best_distance =bruteforce_complete_graph_auxiliary_function(new_vertices, i,
current_distance, graph)

    return best_distance

#

def bruteforce_complete_graph(graph, initial_vertex):

    global best_distance
    best_distance = infinity
    remaining_vertices = list(range(len(graph)))
    remaining_vertices.remove(initial_vertex)
    bruteforce_complete_graph_auxiliary_function(remaining_vertices, initial_vertex, 0, graph)
    return best_distance
```

This post is visible to everyone.

Add a Response

1 response

[Ateesh](#)

7 months ago



Result come as: the distance needed to explore all pieces of cheese using greedy algorithm is: 87 the distance needed to explore all pieces of cheese using bruteforce is: 94

Please advice



This is a mistake that we have seen quite often in the discussions. You should not use **current_distance = current_distance+ graph[current_vertex][i]** because you change the value for each time you go through the loop. Consider calling **best_distance =bruteforce_complete_graph_auxiliary_function(new_vertices, i, current_distance+ graph[current_vertex][i], graph)** instead.

posted 7 months ago by [vgripon](#) (Staff)

Add a comment

Showing all responses

Add a response:

Preview

Submit



edX

[About](#)

[edX for Business](#)

Legal

[Terms of Service & Honor Code](#)

[Privacy Policy](#)

[Accessibility Policy](#)

Connect

[Blog](#)

[Contact Us](#)

[Help Center](#)



© 2020 edX Inc. All rights reserved.

| 深圳市恒宇博科技有限公司 [粤ICP备17044299号-2](#)