

a star algorithm - C:/Users/DRER/Desktop/a star algorithm (3.10.5)

File Edit Format Run Options Window Help

def aStarAlgo(start\_node, stop\_node):

```
    open_set = set(start_node)
    closed_set = set()
    g = {} #store distance from starting node
    parents = {} # parents contains an adjacency map c

    #distance of starting node from itself is zero
    g[start_node] = 0
    #start_node is root node i.e it has no parent node
    #so start_node is set to its own parent node
    parents[start_node] = start_node
```

```
    while len(open_set) > 0:
        n = None

        #node with lowest f() is found
        for v in open_set:
            if n == None or g[v] + heuristic(v) < g[n]:
                n = v
```

```
    if n == stop_node or Graph_nodes[n] == None:
        pass
    else:
        for (m, weight) in get_neighbors(n):
            #nodes 'm' not in first and last set
            #n is set its parent
            if m not in open_set and m not in closed_set:
                open_set.add(m)
                parents[m] = n
                g[m] = g[n] + weight
```

```
    #for each node m,compare its distance
```

IDLE Shell 3.10.5

File Edit Shell Debug Options Window Help

Python 3.10.5 (tags/v3.10.5:ef377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.

```
>>> ===== RESTART: C:/Users/DRER/Desktop/a star algorithm =====
Path found: ['A', 'E', 'D', 'G']
>>>
```

3:55

0.00 KB/S VoLTE 4G 90



Geethika

Today, 10:42 am



```
a star algorithm - C:/Users/DRKR/Desktop/a star algorithm (3.10.5)
File Edit Format Run Options Window Help
Open_set.remove(n)
closed_set.add(n)

print('Path does not exist!')
return None

#define function to return neighbor and its distance
#from the passed node
def get_neighbors(v):
    if v in Graph_nodes:
        return Graph_nodes[v]
    else:
        return None
#for simplicity we'll consider heuristic distances given
#and this function returns heuristic distance for all nodes
def heuristic(n):
    H_dist = {
        'A': 11,
        'B': 6,
        'C': 99,
        'D': 1,
        'E': 7,
        'G': 0,
    }

    return H_dist[n]

#Describe your graph here
Graph_nodes = {
    'A': [('B', 2), ('E', 3)],
    'B': [('C', 1), ('G', 9)],
    'C': None,
    'E': [('D', 6)],
    'D': [('G', 1)],
}
```

```
IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/DRKR/Desktop/a star algorithm =====
>>> Path found: ['A', 'E', 'D', 'G']
```