import heapq

graph = {}

heuristic = {}

cost = {}

st = input("Start node: ").lower()

destination = 'bucharest'

out = open("output.txt", "w")

with open("23341109\_Jafor\_CSE422\_02\_Assignment01\_Summer2024\_InputFile.txt") as inputFile:

  for line in inputFile:

    lst = line.split()

    heuristic[lst[0].lower()] = int(lst[1])

    cost[lst[0].lower()] = [float('inf'), float('inf')]

    graph[lst[0].lower()] = []

    for i in range(2, len(lst), 2):

      graph[lst[0].lower()].append((int(lst[i+1]), lst[i].lower()))

cost[st] = [0, 0 + heuristic[st]]

pq = [(cost[st][1], st)]

v = len(graph)

parent = {}

while pq:

  c, u = heapq.heappop(pq)

  for w, v in graph[u]:

    #print(w, v)

    if cost[v][1] > cost[u][0]+w+heuristic[v]:

      cost[v][0] = cost[u][0] + w

      cost[v][1] = cost[u][0] + w + heuristic[v]

      heapq.heappush(pq, (cost[v][1], v))

      parent[v] = u

  if u == destination:

    break

if cost[destination][0] == float('inf'):

  print("NO PATH FOUND")

else:

  temp = destination

  path = ""

  while temp != st:

    t = temp[0].upper()

    city = t+temp[1:]

    path = "->"+city+path

    temp = parent[temp]

  t = temp[0].upper()

  temp = t+temp[1:]

  path = temp + path

  out.write("Path: "+path+"\n")

  out.write("Total distance: "+str(cost[destination][0])+" km")

inputFile.close()

out.close()