|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Design & Analysis of Algorithms** | **Course Code:** | **CS2009** |
| **Program:** | **BS (Computer Science)** | **Semester:** | **Spring 2023** |
| **Duration:** | **15 Minutes** | **Total Marks:** | **10** |
| **Paper Date:** | **30-March-2023** | **Weight:** | **4** |
| **Section:** | **J** | **Page(s):** | **1** |
| **Exam:** | **Quiz 3** | **Reg. No.** |  |
| **Instruction/Notes:** |  | | | |

**Question 1: CLO 1, [10 marks]**

You are playing Mission Critical and you are stuck on level 9999. To progress to level 10000, you need to complete a real-life mission where you have to save a kid who has been tied to the top of Minar-e-Pakistan. You are currently at Wagah Border. To reach Minar-e-Pakistan, in time, you need to travel at superhuman speed. There is a special ‘Booster Shot’ that helps you travel at superhuman speeds but one ‘Booster Shot’ takes you up to k blocks at maximum. Also, you can only take one ‘Booster Shot’ at a time. There are ‘n’ clinics C1 < C2 < C3 < ……. < Cn along the way, where you can get a ‘Booster Shot’. Clinic Ci is i blocks away from your start location and distance between two clinics is k blocks at maximum. Your task is to get as few shots as possible, along the way, so that you can reach in time to save the kid. Give the most efficient greedy algorithm for this problem to determine which stops you should make for the shot.

**Solution:**

current\_location = start\_location

stops = []

i = 1

remaining\_distance = distance\_to\_destination

while remaining\_distance > 0:

next\_stop = None

max\_distance = 0

for each clinic C[i] in the list of clinics:

if C[i] <= current\_location + k:

distance = C[i] - current\_location

if distance > max\_distance:

max\_distance = distance

next\_stop = C[i]

if next\_stop is not None:

stops.append(next\_stop)

current\_location = next\_stop

remaining\_distance = distance\_to\_destination - current\_location

i = 1

else:

i = i + 1

return stops