C950: Tasks

A : ALGORITHM SELECTION

*Identify by name the self-adjusting algorithm used to create a program to deliver the packages and meet all requirements specified in the scenario.*

The program will use a two opt algorithm to calculate the best tour from a given list of addresses. It adjusts depending on the addresses input as well as their associated weights. It will also include a function that evaluates whether the tour meets all package deadlines.

B1: LOGIC COMMENTS

*The submission accurately explains the algorithm’s logic using pseudocode.*

ARGS: tour[0, 1, 2, 3, 4], distanceMatrix[5][5]

GET length of tour

GET distance of tour by using distanceMatrix

SET improved to True

WHILE tour is improved

FOR each number in tour (i)

FOR each number in tour +1 (j)

CREATE new tour by swapping numbers at i and j

e.g. (0,1,2,3,4)->(0,2,1,3,4)->(0,3,2,1,4)

GET distance of new tour

IF new distance is more than current distance THEN

UPDATE tour equal to new tour

UPDATE current distance equal to new distance

UPDATE improved to true

CONTINUE WHILE

ELSE continue

ENDWHILE

B2: DEVELOPMENT ENVIRONMENT

*The submission accurately describes the software and hardware used to create the Python application.*

Programming Language: Python 3.12.4

Text Editor: VS Code version 1.91.1

OS: Windows 10 Home, version 22H2

Hardware: Dell XPS 13, Processor: Intel i7-8550U

B3: SPACE-TIME AND BIG-O

*The (code) submission accurately describes the space-time complexity for each major block of code and the entire program using Big-O notation.*

Additional details about these and other functions can be found commented within the code

|  |  |  |  |
| --- | --- | --- | --- |
| Function | File | Time Complexity | Space Complexity |
| updatePkgsStatus() | Main.py | O(m\*n) | O(m\*n) |
| getPackageDataList() | Package.py | O(n) | O(n) |
| organizePackages() | Packages.py | O(n^3) | O(n) |
| matrixAttributes() | DistanceMatrix.py | O(r\*(v+e)) | O(v^2) |
| bestTour() | DistanceMatrix.py | O(m\*n^3) | O(v+n) |
| populateTable() | HashTable.py | O(2n^2) | O(2n^2) |
| Insert(), resize() | HashTable.py | O(2n) | O(2n) |
| Update(), find() | HashTable.py | O(n) | O(n) |
| orderPackagesByRoute() | Truck.py | O(m\*n) | O(n) |

B4: ADAPTABILITY

*The submission application accurately explains the application’s capability to scale and adapt to an increasing number of packages.*