Laboratory practice No. IV: Greedy algorithms

Carla Daniela Rendón
Universidad Eafit
Medellín, Colombia
cdrendonb@eafit.edu.co

Juan Diego Gutierrez
Universidad Eafit
Medellín, Colombia
Correointegrante2@eafit.edu.co

3) Practice for final project defense presentation

1. A greedy algorithm is an algorithmic paradigm that follows the problem solving heuristic of making the locally optimal choice at each stage with the intent of finding a global optimum. In many problems, a greedy strategy does not usually produce an optimal solution, but nonetheless a greedy heuristic may yield locally optimal solutions that approximate a globally optimal solution in a reasonable amount of time. The Travelling Salesman Problem (often called TSP) is a classic algorithmic problem in the field of computer science and operations research. It is focused on optimization. In this context better solution often means a solution that is cheaper. TSP is a mathematical problem. It is most easily expressed as a graph describing the locations of a set of nodes. A greedy strategy for the traveling salesman problem (which is of a high computational complexity) is the following heuristic: "At each step of the journey, visit the nearest unvisited city."
2. When we solve the problem of the traveler agent using greedy algorithms, it is not guaranteed that the given solution is the best. In general, the process followed is as follows. Step 1: The distance starts at 0 "zero" and an empty neighborhood arrangement. Step 2: Take an initial node, usually the first node of the graph. Step 3: The distances are calculated with all your neighbors that are not on the visited list. Step 4: Choose the nearest neighbor to the initial node. Step 5: The distance is updated with the distance accumulated plus the distance of the nearest neighbor. Step 6: The initial node is saved in the visited list, the selected neighbor is now the initial node. Step 7: Repeat from step 2 until the destination is reached.
   The algorithm takes a node and visits its neighbors to calculate the distances, chooses the lowest cost and repeats the process. In a totally connected graph and calculate all the neighbors that have not been visited before, choosing at the lowest cost. The problem is smaller in each iteration until reaching the destination. Therefore: θ n log n, where n are cities.
3. The problem to solve is the following: In a city there are n bus drivers. There are also n bus routes in the morning and n bus routes in the afternoon with several durations. Each driver is assigned a route in the morning and a route at night. For each driver, if the total duration of his route for one day exceeds d, he must be paid for the extra time for each hour after its basic at a rate of r pesos per hour. we must assign a route in the morning and a route in the afternoon to each driver, the total extra time that must be paid by the company must be the minimum. We will solve this problem by using the greedy algorithm

4) Practice for midterms

1. temp = actividades[j];
   actividades[j] = actividades[j+1];
   actividades[j+1] = temp;
2. min > adjacencyMatrix[element][i]
3. *a) temp/2 b)temp+ min c) O(i)*

PROFESSOR MAURICIO TORO BERMÚDEZ
Phone: (+57) (4) 261 95 00 Ext. 9473. Office: 19 - 627
E-mail: mtorobe@eafit.edu.co