Assignment 2

Combinatorial Optimization EBM112A05

Submit by December 16, 2014; 17:00 hrs

Note:

1. *Both the problems in this assignment are compulsory. Problem 1 carries 4 points and Problem 2 carries 6 points.*
2. *Please mention the names of all group members in your submission.*

Problem 1: The president's office at a university needs to assign a targeted group of n faculty members to be chairs of n committees. Each person proposes, in decreasing order of preference, a list of three committees that he or she would like to chair. We want to determine whether we could possibly find a satisfiable assignment (i.e., one that assigns the faculty members to the committees so that each faculty member obtains a job on his or her list). If some satisfiable assignment is possible, we want to find the assignment that (a) maximizes the number of faculty members with their most preferred committee chair, and further, (b) among assignments that are not the corresponding faculty members' first choice, the assignment that maximizes the number of faculty members with their second most preferred committee chair. Show how to solve this problem by solving a single assignment problem. Implement your solution to the 20-faculty problem given in the data file `chair.txt`. The format of the file is

---

```
<faculty F 1> <F 1's choice 1> <F 1's choice 2> <F 1's choice 3>
...
<faculty F 20> <F 20's choice 1> <F 20's choice 2> <F 20's choice 3>
```

Problem 2: Choose either Part (a) or Part (b) for this problem but not both.

Part (a)

Perform a computational comparison of methods to solve the binary knapsack problem and submit a report (not more than three pages including tables). First compare the performance of two branch and bound algorithms, one using the linear programming relaxation based bound and the other using the Martello and Toth bound. Your comparison should be based on the number of subproblems they generate during their execution.

Then compare the execution times required by these algorithms with that required by the Dynamic Programming algorithm.

Finally compare the quality of solutions output by an optimal algorithm and the GREEDY and GREEDYFIXED heuristics.

Your test will be based on ten problems. To generate these problems use the `kpgen.exe` file provided and follow the instructions given. (In your report please mention the student code that you used to generate the problems.) The format of the problems generated is

---

```
<size of the problem n>
<profit 1> <profit 2> ... <profit n>
<weight 1> <weight 2> ... <weight n>
```

Part (b)

Devise a heuristic for the binary knapsack problem which has a worst case performance ratio of 0.75. Provide a pseudocode for the algorithm, and prove that its worst case performance ratio is indeed 0.75. Present a computational comparison of the solutions output by this heuristic and the optimal solution on ten problems. To generate these problems use the `kpgen.exe` file provided and follow the instructions given. (In your report please mention the student code that you used to generate the problems.) The format of the problems generated is

---

```
<size of the problem n>
<profit 1> <profit 2> ... <profit n>
<weight 1> <weight 2> ... <weight n>
```

If required, refer to O.H. Ibarra and C.E. Kim. "Fast Approximation Algorithms for the Knapsack and Sum of Subset Problems." Journal of the ACM, Volume 22 Issue 4, Oct. 1975, pages 463–468. (Also remember that you can obtain an optimal solution to the binary knapsack problem using dynamic programming.)

Good luck!