Solutions to Midterm 2 for CS 170

Problem 1. [Greedy] (30 points)

Give the pseudo-code for a Greedy algorithm that solves the following optimization problem. Justify briefly why your algorithm finds the optimum solution. What is the asymptotic running time of your algorithm in terms of n?

There are n gas stations $S_1, \ldots, S_n$ along I-80 from San Francisco to New York. On a full tank of gas your car goes for D miles. Gas station S_1 is in San Francisco, each gas station S_i, for $2 \leq i \leq n$, is $d_i \leq D$ miles after the previous gas station S_{i-1}, and gas station S_n is in New York. What is the minimum number of gas stops you must make when driving from San Francisco to New York?

Answer: Always get gas as late as possible:

```
d := 0; k := 1;
for i := 1 to n − 1 do
      if d + d_{i+1} > D then d := 0; k := k + 1 fi;
      d := d + d_{i+1}
      od;
return k.
```

Suppose this algorithm returns k gas stops, but the optimum is to refuel at the gas stations $S_{i_1}, S_{i_2}, \ldots, S_{i_m}$ for $m < k$. Let S_{i_j} be the first of these gas stations that is later than the j-th stop recommended by the algorithm (such a j must exist because otherwise $m \geq k$). Then your car would run out of gas between $S_{i_{j-1}}$ and S_{i_j}, which is a contradiction.

The running time is $O(n)$.

Problem 2. [Dynamic Programming] (45 points)

Give the pseudo-code for a dynamic-programming algorithm that solves the following optimization problem. What is the recursive relationship between the optimal solutions for subproblems that your algorithm exploits? What is the asymptotic running time of your algorithm in terms of n?

A paragraph consisting of the n words $W_1, \ldots, W_n$ (in this order) is to be broken into lines without breaking any word. The length of a line is L (possibly fractional) inches. The length of each word W_i is $w_i \leq L$ inches. The minimum length of the blank space between adjacent words on a line is u inches. If a line contains more than one word, then all blank spaces between words are "stretched" equally so that the last word of the line ends at the length of the line. If the stretched blank spaces of a line have length $x \geq u$, then the penalty *of the line is $x - u$. In particular, if a line contains the $k \geq 2$ words $W_i, W_{i+1}, \ldots, W_{i+k-1}$, then the penalty of the line is $((L - w_i - w_{i+1} - \cdots - w_{i+k-1})/(k-1)) - u$. In the special case* *that a line contains only a single word W_i, the penalty of the line is $L - w_i$. The cost of the paragraph is the sum of all line penalties. What is the minimum cost of the paragraph?*

Answer: Let $p[i, i+k]$ be the minimum cost of the paragraph consisting of the words $W_i, \ldots, W_{i+k}$. If $k = 0$, then $p[i, i+k] = L - w_i$; if $k \geq 1$ and $w_i + \cdots + w_{i+k} + k \cdot u \leq L$, then $p[i, i+k] := ((L - w_i - \cdots - w_{i+k})/k) - u$; otherwise $p[i, i+k] := \min_{0 \leq j < k}(p[i, i+j] + p[i+j+1, i+k])$. Hence:

```
for i := 1 to n do p[i, i] := L − w_i od;
for k := 1 to n − 1 do
    for i := 1 to n − k do
        if w_i + ··· + w_{i+k} + k · u ≤ L
            then p[i, i + k] := ((L − w_i − ··· − w_{i+k})/k) − u
            else p[i, i + k] := p[i, i];
                for j := 1 to k − 1 do
                    if p[i, i + k] > p[i, i + j] + p[i + j + 1, i + k] then
                        p[i, i + k] := p[i, i + j] + p[i + j + 1, i + k]
                        fi
                    od
            fi
        od
    od;
return p[1, n].
```

The running time is $O(n^3)$.

Problem 3. [Linear Programming] (30 points)

Formulate the following optimization problem as a linear program. Explain what each variable means.

The m generators $G_1, \ldots, G_m$ supply the n cities $C_1, \ldots, C_n$ with power. Each generator G_i can produce at most g_i megawatts per day. Each city C_j consumes c_j megawatts per day. The cost of producing 1 megawatt at generator G_i and shipping it to city C_j is p_{ij} dollars. Each city is legally obligated to purchase at most half of its power from any one generator. What is the minimum total daily cost of power for the cities?

Answer: Minimize $\Sigma_{i=1}^{m} \Sigma_{j=1}^{n} p_{ij} \cdot x_{ij}$ subject to

$\Sigma_{j=1}^{n} x_{ij} \leq g_i$ for all $1 \leq i \leq m$,
$\Sigma_{i=1}^{m} x_{ij} = c_j$ for all $1 \leq j \leq n$,
$2x_{ij} \leq c_j$ for all $1 \leq i \leq m$ and $1 \leq j \leq n$.

Each variable x_{ij} stands for the megawatts shipped from generator G_i to city C_j.

Problem 4. [Network Flow] (45 points)

Formulate the following optimization problem as a max-flow problem. What is the asymptotic running time of the Edmonds-Karp algorithm on your network in terms of m, n, and k?

There are m houses $H_1, \ldots, H_m$ for sale, n persons $B_1, \ldots, B_n$ looking to buy a house, and k real-estate agents $A_1, \ldots, A_k$. Each agent A_i knows a subset $h_i \subseteq \{H_1, \ldots, H_m\}$ of the properties and a subset $b_i \subseteq \{B_1, \ldots, B_n\}$ of the potential buyers. Due to workload constraints, each agent A_i can close at most a_i real-estate transactions. What is the maximum total number of transactions that can be arranged?

Answer: Construct a network with the nodes s (source), $H_1, \ldots, H_m$, $A_1, \ldots, A_k$, $A'_1, \ldots, A'_k$, $B_1, \ldots, B_n$, and t (sink). Note that there are two nodes for each agent. There is an edge of capacity 1 from s to each H_j node. There is an edge of capacity 1 from H_j to A_i iff $H_j \in h_i$. There is an edge from A_i to A'_i of capacity a_i. There is an edge of capacity 1 from A_i to B_j iff $B_j \in b_i$. There is an edge of capacity 1 from each B_j node to t.

The running time of Edmonds-Karp is $O(|V| \cdot |E^2|) = O((m + n + k) \cdot (m + n) \cdot k)$.