CSCI3160: Regular Exercise Set 6

Prepared by Yufei Tao

Problem 1*. Let A be an array of n integers. Define a function $f(x)$ — where $x \geq 0$ is an integer — as follows:

$$f(x) = \begin{cases} 0 & \text{if } x = 0 \\ \max_{i=1}^{x}(A[i] + f(x-i)) & \text{otherwise} \end{cases}$$

Consider the following algorithm for calculating $f(x)$:

algorithm $f(x)$
if $x = 0$ **then return** 0
$max = -\infty$
for $i = 1$ **to** x
    $v = A[i] + f(x-i)$
    if $v > max$ **then** $max = v$
return max

Prove: the above algorithm takes $\Omega(2^n)$ time to calculate $f(n)$.

Problem 2 (The Piggyback Technique). Recall that, for the rot cutting problem, we derived the function $opt(n)$ — the optimal revenue from cutting up a rod of length n — as follows:

$$\begin{aligned} opt(0) &= 0 \\ opt(n) &= \max_{i=1}^{n} P[i] + opt(n-i) \end{aligned} \tag{1}$$

For $n \geq 1$, define $bestSub(n) = k$ if the maximization in (1) is obtained at $i = k$. Answer the following questions:

- Explain how to compute $bestSub(t)$ for all $t \in [1, n]$ in $O(n^2)$ time.
- Assume that $bestSub(t)$ has been computed for all $t \in [1, n]$. Explain how to output an optimal way to cut the rod in $O(n)$ time.

Problem 3*. Let A be an array of n integers. Define function $f(a, b)$ — where $a \in [1, n]$ and $b \in [1, n]$ — as follows:

$$f(a,b) = \begin{cases} 0 & \text{if } a \geq b \\ (\sum_{c=a}^{b} A[c]) + \min_{c=a+1}^{b-1}\{f(a,c) + f(c,b)\} & \text{otherwise} \end{cases}$$

Design an algorithm to calculate $f(1, n)$ in $O(n^3)$ time.

Problem 4. In Lecture Notes 8, our algorithm for computing $f(n, m)$ has space complexity $O(nm)$, i.e., it uses $O(nm)$ memory cells. Reduce the space complexity to $O(n + m)$.

Problem 5*. Let $G = (V, E)$ be a directed acyclic graph (DAG). For each vertex $u \in V$, let $\text{IN}(u)$ be the set of in-neighbors of u (recall that a vertex v is an in-neighbor of u if E has an edge from v to u). Define function $f : V \to \mathbb{N}$ as follows:

$$f(u) = \begin{cases} 0 & \text{if } \text{IN}(u) = \emptyset \\ 1 + \min_{v \,\in\, \text{IN}(u)} f(v) & \text{otherwise} \end{cases}$$

Design an algorithm to calculate $f(u)$ of every $u \in V$. Your algorithm should run in $O(|V| + |E|)$ time. You can assume that the vertices in V are represented as integers $1, 2, ..., |V|$.

Problem 6.** Let $G = (V, E)$ be a directed acyclic graph (DAG). Design an algorithm to find the length of the longest path in G (recall that the length of a path is the number of edges in the path). Your algorithm should run in $O(|V| + |E|)$ time. You can assume that the vertices in V are represented as integers $1, 2, ..., |V|$.