Design and Analysis of Algorithms (Problem Set 4)

November 7, 2024

1 Amortized Analysis and Fibonacci Heaps

1. Show how to implement a queue with two ordinary stacks so that amortized cost of each Enqueue and each Dequeue operation is $O(1)$.

2. In the KMP algorithm for pattern matching, we assumed that the π function associated with the pattern $P[1 \dots m]$ was available. You have to design $O(m)$ time algorithm for computing π.

   Execute your algorithm to compute the π function for the pattern *ababbabbabbababbabb*.

   Hint: There is a great amount of similarities between the π function and Computing $F(i)$ function discussed in the class.

3. A multistack consists of an infinite series of stacks $S_0, S_1, S_2, \dots$, where the ith stack S_i can hold up to 3^i elements. Whenever a user attempts to push an element onto any full stack S_i, we first pop all the elements off S_i and push them onto stack S_{i+1} to make room. (Thus, if S_{i+1} is already full, we first recursively move all its members to S_{i+2}.) Moving a single element from one stack to the next takes $O(1)$ time.

   (a) In the worst case, how long does it take to push one more element onto a multi-stack containing n elements?

   (b) Prove that the amortized cost of a push operation is $O(\log n)$, where n is the maximum number of elements in the multi-stack.

4. Give a linear time algorithm to determine whether a text T is a cyclic rotation of another string T'. For example, *arc* and *car* are cyclic rotations of each other.

5. Write a neat pseudocode for the DECREASE-KEY(H, x) in a Fibonacci Heap?

6. Design an efficient algorithm for deleting an element from a Fibonacci Heap. The amortized cost must be $O(\log n)$.

7. Let v be any node in a Fibonacci heap. We showed that if the size of the subtree rooted at v is m, then the degree of v is $O(\log m)$. Can we say the same thing about the height as well? That is, will the height of v be bounded by $O(\log m)$? Note that all operations, including merging of Fibonacci heaps is allowed.

   Hint: There exists a sequence of operations that may result in a Fibonacci heap which will be a single tree that is just a vertical chain of m elements. Invent one such sequence.

2 NP-Completeness

1. Let A and B be any two computational problems. Let χ be any algorithm for solving B. Problem A is said to be reducible to problem B in polynomial time if each instance I of A can be solved by

   - a polynomial number of executions of χ on instances (of B) each of which are also polynomial of size of I,
   - and, if required, basic computational steps (each taking $O(1)$ time) which are also polynomial in the size of I.

   Convince yourself that this definition of $\leq_p$ subsumes the definition of polynomial time reducibility discussed in the class.

2. Let problem A be defined as follows. Given any undirected graph and an integer k, determine if the graph has an independent set of size at least k.

   Let problem B be defined as follows. Given any undirected graph and an integer t, determine if the graph has a vertex cover of size k. Using the definition of $\leq_p$ given in the previous exercise, show that $A \leq_p B$.

3. For each of the two questions below, decide whether the answer is (i) yes, (ii) no, (iii) unknown, because it would resolve the question of whether "P=NP". Give a brief explanation of your answer.

   (a) Let us define the decision version of the Interval Scheduling Problem (discussed under the topic of Greedy algorithms) as follows: IntervalScheduling: Given a collection of Intervals on a time-line, and an integer k, does the collection contain a subset of nonoverlapping intervals of size at least k? Question: Is it the case that IntervalScheduling $\leq_p$ VertexCover?

   (b) Question: Is it the case that IndependentSet $\leq_p$ IntervalScheduling ?

4. Given an undirected graph $G = (V, E)$, a feedback set is a set $X \subseteq V$ with the property that $G - X$ has no cycle. The UndirectedFeedbackSet problem asks: Given G and k, does there exist a feedback set of size at most k? Prove that UndirectedFeedbackSet is NP-complete.

5. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs. G is said to be isomorphic to G' if we can obtain G' from G by renaming its vertices suitably. In formal words, it means the following.

   A 1-1 and onto function $f : V \rightarrow V'$ is said to be an isomorphism if for each pair of vertices $u, v \in V$, $(u, v) \in E$ if and only if $(f(u), f(v)) \in E'$.

   SubgraphIsomorphism problem is defined as follows: Given any two graphs $G = (V, E)$ and $G' = (V', E')$, does there exist any subgraph of G which is isomorphic to G'. Show that SubgraphIsomorphism problem is NP-complete.

6. A clique is a complete graph (edge exists between each pair of its vertices). Consider the Clique problem: Given an undirected graph $G = (V, E)$ and an integer k, does G contain a clique of size k? Show that Clique is NP-complete.

   Hint: Use the fact that IndependentSet is NP-complete.

7. Recall the algorithm for computing vertex cover of a given graph as discussed in class. Prove that the algorithm computes a vertex cover whose size is at most twice the size of minimum-size vertex cover.

   Hint: For each edge picked during the algorithm, at least one of its endpoints must be in the optimal vertex cover.

   In this course we discussed bipartite-matching problem. The notion of matching can be extended naturally to any arbitrary undirected graph. Based on the algorithm, what relationship can you draw between the matching of a graph and a vertex cover of the same graph?