University of Padova
Master's degree in Computer Science

Advanced Algorithms

Spring 2023

February 2, 2024 – 15:00–17:00

First Part: Theory Questions

Question 1 (4 points) Consider the Union-Find data structure, with the Union operation implemented with *union-by-size*. Show the result of the following sequence of Union operations, assuming that the universe of objects is the integers $0-8$ and that ties are broken by making the representative of the second argument point to the representative of the first argument. You are to draw the final set of directed trees of the *parent graph*.

1. initialize($\{0, 1, 2, 3, 4, 5, 6, 7, 8\}$)
2. union(1, 2)
3. union(3, 4)
4. union(4, 5)
5. union(6, 7)
6. union(6, 8)
7. union(2, 4)

Question 2 (4 points) For each of the following problems, say whether it is NP-hard or not and, if not, specify the complexity of the best algorithm seen in class.

(a) Maximum independent set

(b) All-pairs shortest paths

(c) Set cover

(d) Graph connectivity

Question 3 (4 points) Define the traveling salesperson problem and briefly describe the 2-approximation algorithm seen in class for the metric version of the problem.

Second Part: Problem Solving

Exercise 1 (9 points) Consider Dijkstra's algorithm seen in class, which returns the lengths of the shortest paths from a source vertex to all other vertices in directed graphs with nonnegative weights:

(a) Explain how to modify Dijkstra's algorithm to return the shortest paths themselves (and not just their lengths).

(b) Consider the following algorithm for finding shortest paths in a directed graph where edges may have negative weights: add the same large constant to each edge weight so that all the weights become nonnegative, then run Dijkstra's algorithm and return the shortest paths. Is this a valid method? Either prove that it works (i.e., the returned shortest paths are shortest paths in the original graph), or give a counterexample.

(c) Now let's switch to minimum spanning trees, and do the same: add the same large constant to each edge weight and then run Prim's algorithm. Either prove that the returned solution is a minimum spanning tree of the original graph, or give a counterexample.

Exercise 2 (11 points) Let S be a set of n distinct positive integers, and let WORK(S) be a procedure which, given input S, returns an integer by performing n^2 operations. Now consider the following randomized algorithm:

```
RAND_REC(S)
  if |S| <= 1 then return 1
  x = WORK(S)
  p = RANDOM(S)
  S1 = {s in S such that s < p}
  S2 = {s in S such that s > p}
  if (|S1| >= |S2|) then
     y = RAND_REC(S1)
  else
     y = RAND_REC(S2)
  return x + y
```

Applying the following Chernoff bound show that the complexity of RAND_REC(S) is $O(n^2 \log n)$ with high probability. (Hint: recall the analysis of randomized QuickSort.)

Theorem 1. *Let $X_1, X_2, \ldots, X_n$ be independent indicator random variables such that $E[X_i] = p_i, 0 < p_i < 1$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then, for $0 < \delta \leq 1$,*

$$\Pr(X < (1-\delta)\mu) < e^{-\mu\delta^2/2}.$$