

MATH182 HOMEWORK #5
DUE July 29, 2020

Note: while you are encouraged to work together on these problems with your classmates, your final work should be written in your own words and not copied verbatim from somewhere else. You need to do at least five (5) of these problems. All problems will be graded, although the score for the homework will be capped at $N := (\text{point value of one problem}) \times 5$ and the homework will be counted out of N total points. Thus doing more problems can only help your homework score. For the programming exercise you should submit the final answer (a number) *and* your program source code.

Exercise 1. Suppose that we are given a directed acyclic graph $G = (V, E)$ with real-valued edge weights with two distinguished vertices s and t . Describe a dynamic programming approach for finding a longest weighted simple path from s to t . What does the subproblem graph look like? What is the efficiency of your algorithm?

Exercise 2. A **palindrome** is a nonempty string over some alphabet that reads the same forward and backward. Examples of palindromes are all strings of length 1, **civic**, **racecar**, and **aibohphobia** (fear of palindromes).

Give an efficient algorithm to find the longest palindrome that is a subsequence of a given input string. For example, given the input **character**, your algorithm should return **carac**. What is the running time of your algorithm?

Exercise 3. Consider the problem of making change for n cents using the fewest number of coins. Assume that each coin's value is an integer.

- (1) Describe a greedy algorithm to make change consisting of quarters, dimes, nickels, and pennies. Prove that your algorithm yields an optimal solution.
- (2) Suppose that the available coins are in the denominations that are powers of c , i.e., the denominations are c^0, c^1, \dots, c^k for some integers $c > 1$ and $k \geq 1$. Show that the greedy algorithm always yields an optimal solution.
- (3) Give a set of coin denominations for which the greedy algorithm does not yield an optimal solution. Your set should include a penny so that there is a solution for every value of n .
- (4) Give an $O(nk)$ -time algorithm that makes change for any set of k different coin denominations, assuming that one of the coins is a penny.

Exercise 4. Let's consider a long, quiet country road with houses scattered very sparsely along it. (We can picture the road as a long line segment, with an eastern endpoint and a western endpoint.) Further, let's suppose the residents of these houses are avid cell phone users. You want to place cell phone base stations at certain points along the road, so that every house is within four miles of one of the base stations.

Give an efficient algorithm that achieves this goal, using as few base stations as possible.

Exercise 5. Given a list of n natural numbers d_1, d_2, \dots, d_n , show how to decide in polynomial time whether there exists an undirected graph $G = (V, E)$ whose node degrees are precisely the numbers d_1, d_2, \dots, d_n . (That is, if $V = \{v_1, \dots, v_n\}$, then the degree of v_i should be exactly d_i .) G should not contain multiple edges between the same pair of nodes, or "loop" edges with both endpoints equal to the same node.

Exercise 6. A depth-first forest classifies the edges of a graph into tree, back, forward, and cross edges. A breadth-first tree can also be used to classify the edges reachable from the source of the search into the same four categories.

- (1) Prove that in a breadth-first search of an undirected graph, the following properties hold:
 - (a) There are no back edges and no forward edges.
 - (b) For each tree edge (u, v) , we have $v.d = u.d + 1$.
 - (c) For each cross edge (u, v) , we have $v.d = u.d$ or $v.d = u.d + 1$.
- (2) Prove that in a breadth-first search of a directed graph, the following properties hold:
 - (a) There are no forward edges.
 - (b) For each tree edge (u, v) , we have $v.d = u.d + 1$.
 - (c) For each cross edge (u, v) , we have $v.d \leq u.d + 1$.
 - (d) For each back edge (u, c) , we have $0 \leq v.d \leq u.d$.

Exercise 7. An **Euler tour** of a strongly connected, directed graph $G = (V, E)$ is a cycle that traverses each edge of G exactly once, although it may visit a vertex more than once.

- (1) Show that G has an Euler tour if and only if $\text{in-degree}(v) = \text{out-degree}(v)$ for each vertex $v \in V$.
- (2) Describe an $O(E)$ -time algorithm to find an Euler tour of G if one exists.

Exercise 8. Let $G = (V, E)$ be an undirected, connected graph whose weight function is $w : E \rightarrow \mathbb{R}$, and suppose that $|E| \geq |V|$ and all edge weights are distinct.

We define a second-best minimum spanning tree as follows. Let \mathcal{T} be the set of all spanning trees of G , and let T' be a minimum spanning tree of G . Then a **second-best minimum spanning tree** is a spanning tree T such that $w(T) = \min_{T'' \in \mathcal{T} \setminus \{T'\}} \{w(T'')\}$.

- (1) Show that the minimum spanning tree is unique, but that the second-best minimum spanning tree need not be unique.
- (2) Let T be the minimum spanning tree of G . Prove that G contains edges $(u, v) \in T$ and $(x, y) \notin T$ such that $T \setminus \{(u, v)\} \cup \{(x, y)\}$ is a second-best minimum spanning tree of G .
- (3) Let T be a spanning tree of G and, for any two vertices $u, v \in V$, let $\max[u, v]$ denote an edge of maximum weight on the unique simple path between u and v in T . Describe an $O(V^2)$ -time algorithm that, given T , computes $\max[u, v]$ for all $u, v \in V$.
- (4) Give an efficient algorithm to compute the second-best minimum spanning tree of G .

Exercise 9 (Programming exercise). All square roots are periodic when written as continued fractions and can be written in the form:

$$\sqrt{N} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

For example, let us consider $\sqrt{23}$:

$$\sqrt{23} = 4 + \sqrt{23} - 4 = 4 + \frac{1}{\frac{1}{\sqrt{23}-4}} = 4 + \frac{1}{1 + \frac{\sqrt{23}-3}{7}}$$

If we continue we would get the following expansion:

$$\sqrt{23} = 4 + \frac{1}{1 + \frac{1}{3 + \frac{1}{1 + \frac{1}{8 + \dots}}}}$$

The process can be summarized as follows:

$$\begin{aligned}
a_0 &= 4, & \frac{1}{\sqrt{23}-4} &= \frac{\sqrt{23}+4}{7} = 1 + \frac{\sqrt{23}-3}{7} \\
a_1 &= 1, & \frac{7}{\sqrt{23}-3} &= \frac{7(\sqrt{23}+3)}{14} = 3 + \frac{\sqrt{23}-3}{2} \\
a_2 &= 3, & \frac{2}{\sqrt{23}-3} &= \frac{2(\sqrt{23}+3)}{14} = 1 + \frac{\sqrt{23}-4}{7} \\
a_3 &= 1, & \frac{7}{\sqrt{23}-4} &= \frac{7(\sqrt{23}+4)}{7} = 8 + \sqrt{23} - 4 \\
a_4 &= 8, & \frac{1}{\sqrt{23}-4} &= \frac{\sqrt{23}+4}{7} = 1 + \frac{\sqrt{23}-3}{7} \\
a_5 &= 1, & \frac{7}{\sqrt{23}-3} &= \frac{7(\sqrt{23}+3)}{14} = 3 + \frac{\sqrt{23}-3}{2} \\
a_6 &= 3, & \frac{2}{\sqrt{23}-3} &= \frac{2(\sqrt{23}+3)}{14} = 1 + \frac{\sqrt{23}-4}{7} \\
a_7 &= 1, & \frac{7}{\sqrt{23}-4} &= \frac{7(\sqrt{23}+4)}{7} = 8 + \sqrt{23} - 4
\end{aligned}$$

It can be seen that the sequence is repeating. For conciseness, we use the notation $\sqrt{23} = [4; (1, 3, 1, 8)]$, to indicate that the block $(1, 3, 1, 8)$ repeats indefinitely.

The first ten continued fraction representations of (irrational) square roots are:

$$\begin{aligned}
\sqrt{2} &= [1; (2)] & \text{period}=1 \\
\sqrt{3} &= [1; (1, 2)] & \text{period}=2 \\
\sqrt{5} &= [2; (4)] & \text{period}=1 \\
\sqrt{6} &= [2; (2, 4)] & \text{period}=2 \\
\sqrt{7} &= [2; (1, 1, 1, 4)] & \text{period}=4 \\
\sqrt{8} &= [2; (1, 4)] & \text{period}=2 \\
\sqrt{10} &= [3; (6)] & \text{period}=1 \\
\sqrt{11} &= [3; (3, 6)] & \text{period}=2 \\
\sqrt{12} &= [3; (2, 6)] & \text{period}=2 \\
\sqrt{13} &= [3; (1, 1, 1, 1, 6)] & \text{period}=5
\end{aligned}$$

Exactly four continued fractions, for $N \leq 13$, have an odd period.

How many continued fractions for $N \leq 10000$ have an odd period?

Exercise 10 (Programming exercise). Consider quadratic Diophantine equations of the form:

$$x^2 - Dy^2 = 1$$

For example, when $D = 13$, the minimal solution in x is $649^2 - 13 \times 180^2 = 1$.

It can be assumed that there are no solutions in positive integers when D is square.

By finding minimal solutions in x for $D = \{2, 3, 5, 6, 7\}$, we obtain the following:

$$3^2 - 2 \times 2^2 = 1$$

$$2^2 - 3 \times 1^2 = 1$$

$$9^2 - 5 \times 4^2 = 1$$

$$5^2 - 6 \times 2^2 = 1$$

$$8^2 - 7 \times 3^2 = 1$$

Hence, by considering minimal solutions in x for $D \leq 7$, the largest x is obtained when $D = 5$.

Find the value of $D \leq 1000$ in minimal solutions of x for which the largest value of x is obtained.

(This link might be helpful: https://en.wikipedia.org/wiki/Pell%27s_equation)

Exercise 11 (Programming exercise). Consider the fraction n/d , where n and d are positive integers. If $n < d$ and $\gcd(n, d) = 1$, it is called a **reduced proper fraction**.

If we list the set of reduced proper fractions for $d \leq 8$ in ascending order of size, we get:

$1/8, 1/7, 1/6, 1/5, 1/4, 2/7, 1/3, 3/8, 2/5, 3/7, 1/2, 4/7, 3/5, 5/8, 2/3, 5/7, 3/4, 4/5, 5/6, 6/7, 7/8$

It can be seen that $2/5$ is the fraction immediately to the left of $3/7$.

By listing the set of reduced proper fractions for $d \leq 1000000$ in ascending order of size, find the numerator of the fraction immediately to the left of $3/7$.

Exercise 12 (Programming exercise). It is possible to write ten as the sum of primes in exactly five different ways:

$$7 + 3$$

$$5 + 5$$

$$5 + 3 + 2$$

$$3 + 3 + 2 + 2$$

$$2 + 2 + 2 + 2 + 2$$

What is the first value which can be written as the sum of primes in over five thousand different ways?

Exercise 13 (Programming exercise). Do Project Euler Problem 83: Path sum: four ways <https://projecteuler.net/problem=83>. As a warmup, you might want to do problems 81 and 82 first.