CS 124 Homework 6: Spring 2021

Your name:

Collaborators:

No. of late days used on previous psets:
No. of late days used after including this pset:

Homework is due Wednesday 2021-04-21 at 11:59pm ET. You are allowed up to **twelve** (college)/**forty** (extension school) late days through the semester, but the number of late days you take on each assignment must be a nonnegative integer at most **two** (college)/**four** (extension school).

Try to make your answers as clear and concise as possible; style will count in your grades. Be sure to read and know the collaboration policy in the course syllabus. Assignments must be submitted in pdf format on Gradescope. If you do assignments by hand, you will need to scan in your results to turn them in.

For all homework problems where you are asked to design or give an algorithm, you must prove the correctness of your algorithm and prove the best upper bound that you can give for the running time. Generally better running times will get better credit; generally exponential time algorithms (unless specifically asked for) will receive no or little credit. You should always write a clear informal description of your algorithm in English. You may also write pseudocode if you feel your informal explanation requires more precision and detail, but keep in mind pseudocode does NOT substitute for an explanation. Answers that consist solely of pseudocode will receive little or not credit. Again, try to make your answers clear and concise.

1. **(15 points)** My (Professor Mitzenmacher's) RSA public key (n, e) is:

   $$(469478487497204305296287390 81, 372674862636792350620645369 73).$$

   Convert the message

   I want an A

   into a number, using ASCII in the natural way. (So for "A b": in ASCII, A = 65, space = 32, and b = 98; translating each number into 8 bits gives "A b" = 010000010010000001100010 in binary.) Encode the message as though you were sending it to me using my RSA key, and write for me the corresponding encoded message in decimal.

2. **(15 points)** We have considered, in the context of a randomized algorithm for 2SAT, a random walk with a completely reflecting boundary at 0—that is, whenever position 0 is reached, with probability 1 we move to position 1 at the next turn. Consider now a random walk with a partially reflecting boundary at 0– whenever position 0 is reached, with probability 1/4 we move to position 1 at the next turn, and with probability 3/4 we stay at 0. Everywhere else the random walk either moves up or down 1, each with probability 1/2.

   Find the expected number of moves to reach n starting from position i using a random walk with a partially reflecting boundary. (You may assume there is a unique solution to this problem, as a function of n and i; you should prove that your solution satisfies the appropriate recurrence.)

3. **(0 points, optional)**[1] You have been given a square plot of land that has been divided into n rows and columns, yielding n^2 square subplots. Some of these subplots have rocky ground and cannot support plant growth, while others have soil and can support growing a palm tree. You would like to plant palm trees on a subset of the

[1] We won't use this question for grades. Try it if you're interested. It may be used for recommendations/TF hiring.

square subplots (at most one palm tree per subplot) so that every row and every column has exactly the same number p of palm trees. Furthermore, you would like to do this so that p is as large as possible. Devise an efficient algorithm to determine how to accomplish this. (You may give the running time in terms of the time to solve a suitable flow problem.)

4. Show how to reduce the following problems to linear programming.

   - **(15 points)** Find the maximum flow from a vertex s to a vertex t in a directed graph with edge capacities—except that, at each vertex, half the flow into the vertex is lost (or kept) at the vertex, and the other half flows out. The goal is to maximize the flow that reaches the destination t.
   - **(15 points)** Find the maximum flow from a vertex s to a vertex t in a directed graph with edge capacities—except that, for each edge e, there is also a fixed cost c_e for each unit of flow through the edge. We need to find the maximum flow with the minimum cost. That is, there may be many possible flows that achieve the maximum flow; if there is more than one such flow, find the one of minimum cost. (Hint: you may need to use more than one linear program!)

5. **(15 points)** Consider the two-player zero-sum game given by the following matrix. (A positive payoff goes to the row player: e.g. if the row player picks the first row and the column player picks the first column, the row player scores 3 and the column player scores -3.)

$$\begin{bmatrix} 3 & 1 & 0 & -4 \\ 6 & -2 & -2 & 0 \\ -3 & 2 & 3 & -3 \\ -7 & 4 & -5 & 7 \end{bmatrix}$$

   - Write down a linear program to determine the row player strategy that maximizes the value of the game to the row player, in terms of the probabilities c_1, c_2, c_3, c_4 that the column player picks column 1, 2, 3, 4. Do the same for the column player.
   - Solve these linear programs, and give the proper strategies for both players. (You'll probably want to find an existing LP solver.)
   - What is the value of the game? That is, should the column player pay the row player to play, or vice versa, and how much should one player pay the other to make the game fair?

6. **(15 points)** For the Harvard Tutoring Club, there are n high school students who need tutors, and n available tutors in the club to tutor them. Each student needs a tutor, and each tutor can work with at most one student. Each high school student looks at the tutors and decides on a subset of them that they would want to work with. Suppose that for any subset S of the students, the collective set of tutors $T(S)$ that they are willing to work with satisfies the condition $|T(S)| \geq |S|$. Prove that there is a way to assign every student a tutor that they are willing to work with. Hint: think of this as a flow problem.