Assignment 3: Chakra Conundrums
Due Date: November 23, 2020, 11:59pm ET

Instructions

1. Typed assignments are preferred (e.g., PDFs created using LaTeX or Word), especially if your handwriting is possibly illegible or if you do not have access to a good quality scanner. Either way, you need to submit a single PDF named "hwk3.pdf" on MarkUS at https://markus.teach.cs.toronto.edu/csc373-2020-09

2. You will receive 20% of the points for a (sub)question when you leave it blank (or cross off any written solution) and write "I do not know how to approach this problem." If you leave it blank but do not write this or a similar statement, you will receive 10%. This does not apply to any bonus (sub)questions.

3. You may receive partial credit for the work that is clearly on the right track. But if your answer is largely irrelevant, you will receive 0 points.

Preface (Important, MUST read!)

You live in a world dominated by a mysterious energy called *chakra*. This energy takes two forms: *physical energy* and *spiritual energy*. People known as *ninja* find ways to harness chakra and use it to develop mystical arts called *jutsu*. Ninja form *squads* to go on missions, where they use jutsu to fight other ninja and evil forces. You are a ninja in the Hidden Algo village; in fact, you are their chief scientist. Your goal is to help your village chief (the *kage*) optimize battle strategies. But when you fail to do so, you may need to demonstrate that it's because the problem was provably difficult and not because you lacked the skills to solve it.

OK, maybe the last paragraph wasn't *that* important, but the next bit is: Throughout this assignment, you can directly use the following common polynomial-time graph algorithms as subroutines: maximum flow, maximum cardinality matching, shortest paths, checking if a directed graph is acyclic, and checking if an undirected graph is bipartite.

Q1 [20 Points] Judicious Use of Jutsu

Your village is going to send out a ninja on an important upcoming mission. Earlier reconnaissance has provided useful intelligence. There are only two jutsu which are effective against the enemy: *shadow clones* and *rasengans*. They use different amounts of physical and spiritual energy and have different effectiveness against the enemy.

- Each shadow clone uses 41 units of physical energy and 27 units of spiritual energy, and has effectiveness 5. (If there are x shadow clones, multiply all three numbers by x.)

- Each rasengan uses 23 units of physical energy and 39 units of spiritual energy, and has effectiveness 8. (If y ransengans are thrown, multiply all three numbers by y.)

- Effectiveness and physical/spiritual energy consumption add up across the two jutsu.

- The ninja assigned to the mission has 450 units of physical energy and 350 units of spiritual energy available.

Your goal is to help maximize the effectiveness given the energy constraints by selecting the right number of shadow clones and rasengans. Suppose that the number of shadow clones and rasengans do not have to be integers; they can be non-negative real numbers.

(a) [10 Points] Write a linear program in its standard form for the problem. No justification needed.

(b) [10 Points] Write the dual of the linear program from part (a). No justification needed.

Note: You do not need to submit answers to parts (c) and (d) below. They are for fun. You can use an online graphing calculator (e.g. desmos.com) or an online LP solver (e.g. online-optimizer.appspot.com) for these problems.

(c) [0 Points] Find the optimal objective values of the primal and dual linear programs and verify that they indeed match, as guaranteed by strong duality.

(d) [0 Points] If we required their variables to take integer values, would their optimal objective values match?

Q2 [20 Points] Celebrations don't Wait for a Fool to Learn.
Congratulations! Thanks to your help, the ninja successfully completed the mission. Your village has organized festivities to celebrate this success. Organizing the festivities involves eight tasks which need to be completed: $T_1, \ldots, T_8$. Each task has a duration (the amount of time it takes to perform the task), and some tasks depend on others (i.e., they can only start after all the tasks they depend on have finished). The dependencies and durations are given below.

Dependencies	Task	Duration (minutes)
-	T_1	10
-	T_2	25
-	T_3	15
T_1, T_2	T_4	7
T_3, T_4	T_5	18
T_5	T_6	12
T_4	T_7	3
T_6, T_7	T_8	14

So, for example, T_1 and T_2 must be completed before T_4 can begin. Each task must be assigned to a single volunteer who will start the task when you tell them to and finish it after the duration specified in the table above. However, you can assume that there are enough volunteers in the Hidden Algo village — as many as you need — for performing tasks in parallel.

Write a linear program (with real-valued variables) to find the minimum amount of time needed to complete all the tasks.

Q3 [25 Points] Squad Formation

Next up, your village is facing another crucial mission. This time, one ninja just won't do! You have to use all the $2n$ ninja that your village has. Each ninja i has a chakra level c_i, which is a *positive integer*. Given the chakra levels of all $2n$ ninja, you want to partition them into squads such that each squad has both the same number of ninja *and* the same total chakra level.

In the following, for proving NP-completeness, you must use a reduction from the PARTITION problem: *Given a multiset $S = \{a_1, \ldots, a_n\}$ of positive integers, does there exist a subset $S' \subseteq S$ such that both S' and $S \setminus S'$ have the same sum?* In both PARTITION and the following problems, assume that duplicate integers are allowed, but all integers must be strictly positive.

(a) [10 Points] Consider the decision problem MANYSQUADS: *Can the ninja be partitioned into n squads of 2 ninja each such that the total chakra level in each squad is the same?* Either provide a polynomial-time algorithm for MANYSQUADS or prove that it is NP-complete.

(b) [15 Points] Consider the decision problem MANYNINJA: *Can the ninja be partitioned into 2 squads of n ninja each such that the total chakra level in each squad is the same?* Either provide a polynomial-time algorithm for MANYNINJA or prove that it is NP-complete.

[Hint: Part (b) is almost like the PARTITION problem, except that we want the two parts to also have the same cardinality. If you are considering a reduction from PARTITION to MANYNINJA, you might want to convert an instance A of PARTITION with n elements to an B instance of MANYN-INJA with $2n$ elements such that a YES answer to A implies a YES answer to B, i.e., any feasible solution S' of A (even one with $|S'| \neq |S \setminus S'|$) can be mapped to a feasible solution of B where the two squads have equal cardinality and equal sum.

One idea may be to just add n zeros to the multiset in A: that way, given any feasible solution S' of A, we can add $n - |S'|$ zeros to S' and $n - |S \setminus S'|$ zeros to $S \setminus S'$; the equal sum is maintained and the two squads now have equal cardinality n. However, adding zeros is not allowed as the problem requires *positive* integers. Also, we need to make sure that a NO answer to A also implies a NO answer to B. Think about addressing the former issue first and then the latter.]

Q4 [25 Points] Squad Sabotage

The squads are now ready for their missions. One of the squads has been assigned a peculiar mission by the kage, which goes as follows. There are n villages in the country and there are one-way roads from some villages to others (i.e. the villages and the one-way roads form a *directed* graph).

The squad is told that a certain villain will be traveling through the villages along some circular route (i.e. along a cycle in the graph). However, it is not known which circular route the villain will take. The squad is a bit intimidated by the mission for there are so many circular routes the villain could take. The squad can send its members to k villages where they would lie in wait for the villain, but what if the villain takes a circular route that doesn't include any of those k villages?

The squad comes to you with the following question: *Are there k villages such that every circular route will include at least one of them?* For the life of you, you can't seem to be able to solve the problem efficiently. To defend yourself, you want to prove that it's because the problem is actually NP-complete.

(a) [10 Points] First, prove that the problem is in NP.

[Hint: Your first instinct may be to provide, as advice, the choice of k villages as well as *every* cycle in the graph. However, since there could be exponentially many cycles in the graph, this advice would not be of polynomial length. You will need to come up with something better.]

(b) [15 Points] Next, prove that the problem is also NP-hard by producing a reduction from the NP-complete VERTEXCOVER problem defined below:

VertexCover:
Input: An *undirected* graph $G = (V, E)$ and a positive integer k.
Question: Does there exist a subset $S \subseteq V$ of $|S| = k$ vertices such that at least one of the endpoints of every edge is in S?