Worth: 10%

1. **[20 marks]**

   An epidemiologist is studying the transmission of the COVID-19 virus in the Greater Toronto Area (GTA). The community is represented by a graph $G = (V, E)$ where vertices represent people in the GTA and edges represent two individuals who have interacted with each other. To study how the virus may have spread, the epidemiologist wants to know if people in the GTA can be divided into at most k groups $G_1, \dots, G_k$ where:

   - Everyone in V is in some group, and no person is in two different groups.
   - Everyone in group G_i has interacted with each other.

   (a) **[5 marks]** Show that the above problem given the graph G and integer k as inputs is in *NP*.

   (b) **[5 marks]** Show that the above problem when $k \leq 2$ is decidable in polynomial time.

   (c) **[10 marks]** Show that the above problem when $k \geq 3$ is *NP*-Complete.

2. **[20 marks]**

   Consider the following DisjointHamiltonianPaths decision problem ("DHP" for short).

   - *Input:* Graph $G = (V, E)$ — G may be directed or undirected.
   - *Output:* Does G contain at least two edge-disjoint Hamiltonian paths?

   (Two paths are said to be *edge-disjoint* if there is no edge that belongs to both paths.)

   (a) **[12 marks]** Write a *detailed* proof that DisjointHamiltonianPaths is *NP*-complete. State what you are doing at each step of your solution: it will be graded on its structure as much as on its content.

   (b) **[8 marks]** Give a precise definition for the DisjointHamiltonianPaths-Search problem. Then, write a *detailed* argument that this problem is polynomial-time self-reducible. Once again, your solution will be graded on its structure as well as its content, so make sure to state what you are doing at each step.

3. **[20 marks]**

   A propositional formula φ is in **3-Positive Conjunctive Normal Form (3-PCNF)** iff it is written in 3-CNF with no negative literal, i.e., all the variables in the formula are positive. For example, $\varphi = (x_1 \vee x_3 \vee x_4) \wedge (x_2 \vee x_3 \vee x_2) \wedge (x_1 \vee x_2 \vee x_3)$ is in 3-PCNF. Obviously φ is satisfiable by setting all the variables to True.

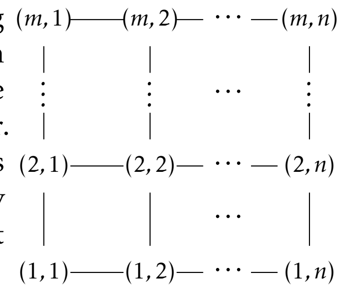
   We are interested in a different question: how *few* variables must be set to True in order to satisfy φ? For example, the previous formula can be satisfied by setting just one variable to True, namely, x_3. This is an NP-Hard problem (you do not need to prove this).

   (a) **[8 marks]** Define the corresponding Decision Problem for this optimization problem, and show that this optimization problem is polynomial-time self-reducible.

   (b) **[12 marks]** Formulate the optimization problem as an integer program. Using this, derive a polynomial-time 3-approximation algorithm for the same. That is, if the minimum number of variables needed to be set to True to make φ satisfiable is k^*, your algorithm should set at most $3k^*$ variables to True. Provide a proof of correctness of your algorithm and compute its time complexity.

4. **[20 marks]**

Your friends want to break into the lucrative coffee shop market by opening a new chain called *The Coffee Pot*. They have a map of the street corners in a neighbourhood of Toronto (shown on the right), and estimates $p_{i,j}$ of the profits they can make if they open a shop on corner (i, j), for each corner. However, municipal regulations forbid them from opening shops on corners $(i-1, j)$, $(i+1, j)$, $(i, j-1)$, and $(i, j+1)$ (for those corners that exist) if they open a shop on corner (i, j). As you can guess, they would like to select street corners where to open shops in order to maximize their profits!



(a) **[5 marks]** Consider the following greedy algorithm to try and select street corners.

> $C := \{(i, j) : 1 \le i \le m, 1 \le j \le n\}$ # C is the set of every available corner
> $S := \emptyset$ # S is the current selection of corners
> **while** $C \ne \emptyset$:
> pick $(i, j) \in C$ with the maximum value of $p_{i,j}$
> # Add (i, j) to the selection and remove it (as well as all corners adjacent to it) from C.
> $S := S \cup \{(i, j)\}$
> $C := C - \{(i, j), (i-1, j), (i+1, j), (i, j-1), (i, j+1)\}$
> **return** S

Give a precise counter-example to show that this greedy algorithm does not always find an optimal solution. State clearly the solution found by the greedy algorithm, and show that it is not optimal by giving another selection with larger profit.

(b) **[15 marks]** Prove that the greedy algorithm from part (a) has an approximation ratio of 4. (Hint: Let S be the selection returned by the greedy algorithm and let T be any other valid selection of street corners. Show that for all $(i, j) \in T$, either $(i, j) \in S$ or there is an adjacent $(i', j') \in S$ with $p_{i',j'} \ge p_{i,j}$. What does this means for all $(i, j) \in S$ and their adjacent corners?)