UBINET, Master 2 IFI Algorithms for telecommunications

**Final Exam, November 2023**

1 hour 30

*No documents are allowed. No computers, cellphones.*

*Instruction and comments: the points awarded for your answer will be based on the correctness of your answer as well as the clarity of the main steps in your reasoning. All proposed solutions must be proved.* **All the exercises are independent.** *The points are indicated so you may adapt your effort.*

**Exercise 1 (Flow. 5 points, 20 minutes)**

1. Give a linear programme that takes an elementary network flow as input (i.e., a digraph $N = (V, A)$ with a source $s \in V$ and a sink $d \in V$ and with arcs'capacities $c : A \to \mathbb{N}$) and computes a maximum $s$-$d$ flow. Precisely describe the variables, the objective function and the constraints.

Let us consider the elementary network flow $N$ depicted in Figure 1.

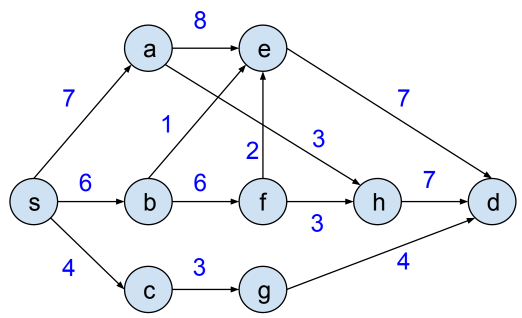

Figure 1: Elementary network $s$-$d$ flow $N$ with arcs' capacity in blue.

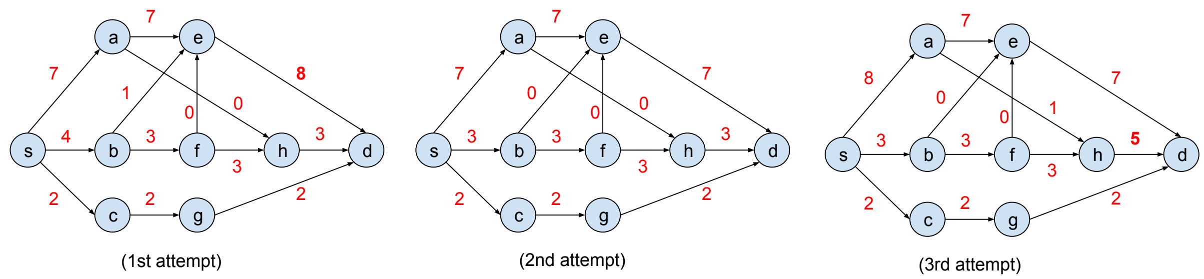


Figure 2: Three attempts of initial $s$-$d$ flows in the elementary network $N$.

2. Among the three pictures in Figure 2, only one represents a $s$-$d$ flow in the elementary network $N$. Which one? Explain.
3. Apply the Ford-Fulkerson Algorithm to $N$ starting from the $s$-$d$ flow $f$ depicted in the middle of Figure 2 (2nd attempt). All steps of the execution of the algorithm must be detailed.

   For each iteration, draw the auxiliary graph, give the chosen path and the amount of flow that you will push, and draw the network with the new flow.
4. What is the final value of the flow? Give a minimum $s$-$d$ cut. Explain why the given $s$-$d$ flow is actually maximum.

**Exercise 2 (Vertex Cover. 8 points, 35 minutes)**

Given a graph $G = (V, E)$, a *vertex cover* is a set $Q \subseteq V$ of vertices such that $Q \cap e \neq \emptyset$ for all $e \in E$, i.e., a set of vertices “touching" all edges of $G$.

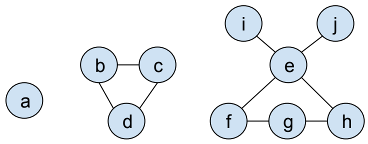


Figure 3: A (not connected) graph $G = (V, E)$

1. Give a vertex cover of size at most 4 for the graph $G$ depicted in Figure 3.
2. Describe (in at most 5 lines) a “real-life" problem that may be modeled by the problem of computing a minimum (of minimum size) vertex cover in a graph.
3. Describe an algorithm that, given a graph $G = (V, E)$ as input, computes a minimum vertex cover of $G$. Give its time-complexity as a function of its number of vertices ($n = |V|$) and edges ($m = |E|$). The description may be by words (at most 5 lines) and/or with a pseudo-code.

   *Hint: You may try all possibilities.*

Let $G = (V, E)$ be a graph and $Q \subseteq V$ be a vertex cover of $G$.

4. Assume that $G$ has a vertex $v \in V$ of degree $> |Q|$ (recall that the *degree* of a vertex is its number of neighbours). Show that $v \in Q$. *Hint: proof by contradiction.*
5. Let $u, v \in V \setminus Q$ be two vertices not in $Q$. Show that $\{u, v\} \notin E$, i.e., there is no edge between $u$ and $v$.
6. If all vertices of $G$ have degree at most $|Q|$, then show that $|E| \leq |Q|^2$.

**Algorithm 1** Algo1$(G, k, Q)$

**Require:** A graph $G = (V, E)$, an integer $k \in \mathbb{N}$ and a set $Q$.

**If** $|E| = 0$, **Return** $Q$
**Else**, **if** $k = 0$, **Return** “*No solution*”
**Else**, **if** there exists a vertex $v \in V$ with degree $> k$
    Let $G'$ be the graph obtained from $G$ by removing $v$ (and its incident edges) and then every isolated (incident to no edges) vertex;
    **Return** $Algo1(G', k - 1, Q \cup \{v\})$
**Else**, **if** $|E| > k^2$, **Return** “*No solution*”
**Else**, do an exhaustive search :
    **If** $G$ admits a vertex cover $Q'$ of size $\leq k$, **Return** $Q \cup Q'$.
    **Else**, **Return** “*No solution*”

7. Apply Algo1$(G, 3, \emptyset)$ for the graph $G$ of Figure 3 (detail all steps). What does it return?
8. Apply Algo1$(G, 4, \emptyset)$ for the graph $G$ of Figure 3 (detail all steps). What does it return?
9. What returns Algo1$(G, k, \emptyset)$ for any graph $G$ and integer $k \in \mathbb{N}$. Explain your answer.
10. What is the time-complexity of $Algo1$ as a function of $n = |V(G)|$ and $k$.

**Exercise 3 (Colouring. 8 points, 35 minutes)**

Given a graph $G = (V, E)$, a *proper $k$-colouring* is a function $c : V \to \{1, \cdots, k\}$ that assigns an integer (a colour) to each vertex such that no two adjacent vertices receive the same colour, i.e., for every $\{u, v\} \in E$, $c(u) \neq c(v)$. The *chromatic number* $\chi(G)$ of a graph $G$ is the minimum $k$ such that $G$ admits a proper $k$-colouring.

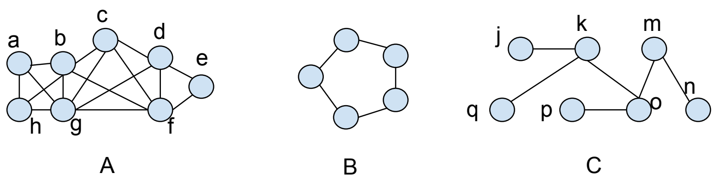


Figure 4: Three connected graphs $A, B$ and $C$.

1. Give a proper 4-colouring to the graph $A$ of Figure 4, a proper 3-colouring for the graph $B$ and a proper 2-colouring for the graph $C$.
2. Show that $\chi(A) = 4$, $\chi(B) = 3$ and $\chi(C) = 2$.
3. Characterize (describe) all graphs $G$ such that $\chi(G) = 1$.

A graph $G = (V, E)$ is *bipartite* if there exists a partition $(A, B)$ of $V$ such that $A$ and $B$ induce stables sets (i.e., no two vertices of $A$ are adjacent, and no two vertices of $B$ are adjacent).

We admit the following theorem

**Theorem 1** *A graph is bipartite if and only if it has no odd cycle.*

4. Show that $\chi(G) \leq 2$ if and only if $G$ is bipartite.
5. Give a linear time algorithm that takes a graph $G$ as input and decides if $G$ is bipartite (prove it).

*Hint: use a BFS algorithm*

A graph $G = (V, E)$ is *$d$-degenerate* if there exists an ordering $(v_1, v_2, \cdots, v_n)$ of its vertices such that, for every $1 \leq i \leq n$, $v_i$ has degree at most $d$ in the graph $G_i = G[v_i, v_{i+1}, \cdots, v_n]$ induced by the vertices $\{v_i, v_{i+1}, \cdots, v_n\}$. In other words, $v_1$ has degree at most $d$ in $G$; once $v_1$ has been removed, then $v_2$ has degree at most $d$ in what remains; once $v_2$ has been removed, $v_3$ has degree at most $d$ in what remains...

6. Show that the graph $A$ of Figure 4 is 3-degenerate and that the graph $C$ of Figure 4 is 1-degenerate.
7. Let $G$ be a $d$-degenerate graph. Show that $\chi(G) \leq d + 1$.

*Hint: by induction on $n = |V(G)|$.*

A graph is *planar* if it can be drawn in the plane without crossing edges.

We admit the following theorems.

**Theorem 2** *Let $G$ be a planar graph. Then, $G$ is 5-degenerate.*

**Theorem 3 (Four-colour Theorem)** *Let $G$ be a planar graph. Then, $\chi(G) \leq 4$.*

8. Give the definition of a 2-approximation algorithm for a minimization problem.
9. Using questions above, give a 2-approximation algorithm for the problem of computing $\chi(G)$ in the class of planar graphs.