

Le barème est indicatif. Vous pouvez utiliser le résultat de toute question, même non traitée.
La rédaction doit être précise. Seules, les réponses correctement justifiées apportent des points.

Exercice 1 : Applications du cours (3 points)

Répondez aux questions suivantes en quelques lignes, en justifiant clairement vos réponses.

1. On considère le langage des codes des machines de Turing qui acceptent exactement les multiples de 42 (codés en base 2). Le langage L est-il décidable? semi-décidable? dans \mathbf{P} ?
2. Tout problème de la classe \mathbf{PSPACE} se réduit-il au problème de correspondance de Post?
3. Si L est un \mathbf{NP} -difficile, est-il vrai que tout langage K contenant L est aussi \mathbf{NP} -difficile?

Exercice 2 : Problèmes \mathbf{NP} -complets, \mathbf{NL} -complets, dans \mathbf{P} ou dans \mathbf{L} (7 points)

Montrer que chacun des problèmes suivants est soit \mathbf{NP} -complet, soit \mathbf{NL} -complet, soit dans \mathbf{P} , soit dans \mathbf{L} , en donnant à chaque fois la plus petite classe possible contenant le problème. Dans tous les problèmes, les entiers donnés en entrée sont codés en binaire.

1. ARBRE DE POIDS FIXÉ

Entrée : Un entier $P \in \mathbb{N}$, un graphe $G = (V, E)$ non orienté, et une fonction $p : E \rightarrow \mathbb{N}$.

Question : Existe-t-il un sous-ensemble $F \subseteq E$ des arêtes de G tel que (V, F) est un arbre (c'est-à-dire un graphe connexe sans cycle) et $\sum_{e \in F} p(e) = P$?

2. CHEMIN QUASI-HAMILTONIEN

Entrée : Un graphe orienté G .

Question : Le graphe G possède-t-il un chemin qui visite chaque sommet au moins une fois et au plus deux fois?

3. FORTE CONNEXITÉ

Entrée : Un graphe **orienté** G .

Question : Le graphe G est-il fortement connexe, c'est-à-dire, est-il vrai que pour tous sommets s et t , il y a un chemin de s à t ?

4. NOT-ALL-EQUAL 4SAT

Entrée : Une formule 4-CNF (c'est-à-dire une conjonction de clauses, chacune du type $(\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4)$, où chaque ℓ_i est un littéral).

Question : Y a-t-il une affectation des variables pour laquelle chaque clause a au moins un littéral vrai et au moins un littéral faux?

5. FACTEUR COMMUN

Entrée : Un alphabet Σ , deux mots $u, v \in \Sigma^*$ et un entier $n \in \mathbb{N}$.

Question : Existe-t-il un mot $w \in \Sigma^*$ de longueur au moins n qui est facteur de u et v (c'est-à-dire que $u = u_1 w u_2$ et $v = v_1 w v_2$)?

6. PARTITION EN 3 CLIQUES

Entrée : Un graphe non orienté $G = (V, E)$.

Question : Existe-t-il une partition de l'ensemble V des sommets de G en trois parties deux à deux disjointes V_1, V_2 et V_3 , avec $V = V_1 \cup V_2 \cup V_3$, telle que pour chaque $i = 1, 2, 3$, le sous-graphe de G construit en gardant les sommets de V_i et les arêtes de G reliant les points de V_i soit une clique (c'est-à-dire que les sommets de V_i sont deux à deux reliés par une arête dans G) ?

Un **semigroupe** est un ensemble S muni d'une multiplication associative $(x, y) \mapsto xy$: pour tous $x, y, z \in S$, on a $(xy)z = x(yz)$. On peut ainsi définir la puissance n -ème d'un élément $s \in S$ sans se préoccuper du parenthésage : s^n désigne la multiplication de n copies de s , c'est-à-dire $\underbrace{ss \cdots s}_{n \text{ fois}}$. On dit qu'un semigroupe S est **42-périodique** si pour tout élément $s \in S$,

il existe un entier $n \geq 1$ tel que $s^n = s^{n+42}$.

7. SEMIGROUPE 42-PÉRIODIQUE

Entrée : Un semigroupe S donné par sa table de multiplication.

Question : Le semigroupe S est-il 42-périodique ?

Exercice 3 : Problèmes P-complets (8 points)

En logique propositionnelle, on appelle *clause de Horn* une clause qui contient **au plus un** littéral positif. Par exemple, les trois clauses suivantes sont des clauses de Horn : (v) , $(\neg v \vee \neg x)$ et $(\neg y \vee \neg z \vee \neg x \vee \neg u \vee v)$. À l'inverse, la clause suivante n'est **pas** une clause de Horn car elle contient deux littéraux positifs (soulignés) : $(\neg y \vee \underline{z} \vee \neg x \vee \neg u \vee \underline{v})$.

1. Montrer que le problème HORN-SAT suivant est dans **P** en décrivant en français un algorithme polynomial.

Entrée : Une conjonction $\varphi = C_1 \wedge C_2 \wedge \cdots \wedge C_n$ de clauses de Horn.

Question : La formule φ est-elle satisfaisable ?

Vous pourrez d'abord considérer le cas où toute clause a au moins un littéral négatif. Sinon, votre algorithme pourra initialiser toutes les variables à « Faux », puis, modifier itérativement cette affectation selon la formule φ , jusqu'à obtenir une conclusion. Justifiez précisément la complexité de l'algorithme obtenu.

2. On se place sur l'alphabet $\{0, 1\}$. Soit $A \subseteq \{0, 1\}^*$ un problème de la classe **P**. On suppose que $A \neq \emptyset$ et $A \neq \{0, 1\}^*$. Quels sont les problèmes de **P** se réduisant polynomialement à A ?

On définit maintenant la notion de **P-complétude**. On dit qu'un problème A est **P-complet** si les deux conditions suivantes sont vérifiées :

(a) $A \in \mathbf{P}$,

(b) tout problème de **P** se réduit à A par une réduction en espace logarithmique.

3. Montrer que le problème HORN-SAT est **P-complet**.

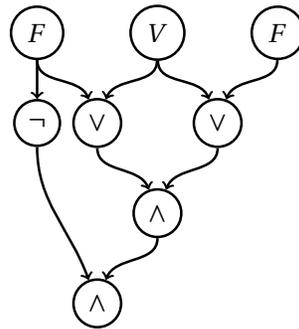
Remarque. Nous n'avons encore rencontré aucun problème **P-complet**. Cela donne une indication sur ce qu'il faut faire dans cette question.

4. Montrer que le problème reste **P-complet** si les clauses de la formule d'entrée contiennent au plus 3 littéraux.

On passe maintenant à un second problème. Un **circuit Booléen** est un **graphe orienté sans circuit** dont les sommets sont appelés des *portes*. Les portes sont étiquetées soit par des valeurs Booléennes, Vrai (V) ou Faux (F), soit par des connecteurs logiques (\wedge pour **Et**, \vee pour **Ou** et \neg pour **Non**). Un circuit doit satisfaire les propriétés suivantes :

- Les portes qui n'ont pas d'arc entrant (appelées *portes d'entrée*) sont étiquetées par une valeur Booléennes : Vrai (V) ou Faux (F).
- Les autres portes sont étiquetées par des connecteurs logiques (\wedge , \vee ou \neg). Une porte étiquetée par \neg a un unique arc entrant, et une porte étiquetée par \wedge ou \vee a exactement 2 arcs entrants.
- Il y a une unique porte sans arc sortant, appelée la *porte résultat*. Les autres portes peuvent avoir un nombre arbitraire d'arcs sortants.

Un exemple de circuit Booléen est donné ci-dessous. La porte résultat, en bas, est étiquetée \wedge .



Chaque porte calcule une valeur Booléenne. Sauf pour la porte résultat, cette valeur est ensuite transmise le long de tous ses arcs qui sortent de la porte. Comme les portes peuvent avoir plusieurs arcs sortants (sauf la porte résultat), une même sortie peut ainsi être utilisée plusieurs fois en entrée. Intuitivement, une porte effectue sur ses arcs entrants l'opération logique correspondant à son étiquette :

- Une porte d'entrée calcule Vrai si son étiquette est V et Faux si son étiquette est F .
- Une porte étiquetée par \neg , et dont l'unique arc entrant transmet la valeur val , calcule $\neg val$.
- Une porte étiquetée par \vee , et dont les deux arcs entrants transmettent des valeurs val_1 et val_2 , calcule $val_1 \vee val_2$.
- Une porte étiquetée par \wedge , et dont les deux arcs entrants transmettent des valeurs val_1 et val_2 , calcule $val_1 \wedge val_2$.

La *valeur calculée par le circuit* est celle de sa porte résultat. On considère le problème ÉVALUATION DE CIRCUIT suivant :

Entrée : Un circuit Booléen C .

Question : Est-ce que le circuit calcule la valeur Booléenne *Vrai*?

5. Montrer que ÉVALUATION DE CIRCUIT est **P**-complet. Pour montrer la **P**-difficulté, vous pouvez utiliser la question 4 même si vous ne l'avez pas faite.
6. On considère maintenant une restriction, appelée ÉVALUATION DE CIRCUIT MONOTONE :

Entrée : Un circuit Booléen C **sans porte étiquetée \neg**

Question : Le circuit C calcule-t-il la valeur *Vrai*?

Montrer que ÉVALUATION DE CIRCUIT MONOTONE est encore **P**-complet.

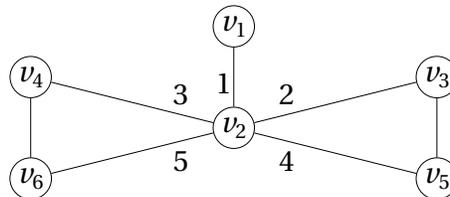
Exercice 4 : Cycles dans les graphes non orientés (5 points)

Dans cet exercice, on considère des graphes **non orientés**. Étant donné un tel graphe G , un *cycle dans G* est une suite finie d'arêtes (e_1, e_2, \dots, e_k) consécutives, telle que pour tout i , $e_i \neq e_{i+1}$, et revenant au sommet de départ. Ainsi, e_i et e_{i+1} partagent exactement un sommet. Notez que si e est une arête, (e, e) ne constitue pas un cycle (à cause de la condition $e_i \neq e_{i+1}$). On considère le problème CYCLE NON ORIENTÉ :

Entrée : Un graphe *non-orienté* G .

Question : Existe-t-il un cycle dans G ?

On veut montrer que CYCLE NON ORIENTÉ est dans **L**. On suppose dans un premier temps qu'avec le graphe d'entrée $G = (V, E)$, on dispose aussi pour chaque sommet $v \in V$ d'une numérotation $1, 2, \dots, d_v$ des arêtes incidentes au sommet v (où d_v est le nombre de telles arêtes). Par exemple, sur le graphe non orienté suivant, les 5 arêtes incidentes à v_2 sont numérotées de 1 à 5. On a une numérotation des arêtes incidentes à chaque autre sommet du graphe, mais sur la figure, cette numérotation n'est représentée que pour v_2 .



On considère un graphe non orienté $G = (V, E)$. Pour chaque sommet $v \in V$ et chaque arête e incidente à v , on définit le parcours (infini) $P(v, e)$ de G suivant. On part du sommet v et on prend l'arête e . Ensuite, à chaque fois qu'on arrive dans un nouveau sommet $u \in V$, si on est entré par l'arête incidente numérotée par k , on prend ensuite celle numérotée par $k + 1$ (si k était l'arête incidente maximale de u , on prend celle numérotée par 1).

On dit que $P(v, e)$ est un *bon parcours* si il repasse par v et que la première fois que cela arrive, l'arête incidente utilisée pour entrer dans v est différente de e .

1. On considère un sommet $v \in V$ et e une arête incidente à v . Montrer que si $P(v, e)$ est un bon parcours, alors G contient un cycle.
2. Montrer que si G contient un cycle, alors il existe un sommet $v \in V$ et une arête $e \in E$ incidente à v tels que $P(v, e)$ est un bon parcours.
3. Dédurre des questions 1 et 2 que le problème CYCLE NON ORIENTÉ, restreint aux entrées pour lesquelles on a une numérotation des arêtes incidentes à chaque sommet, est dans **L**.
4. Expliquer pourquoi l'hypothèse qu'on dispose d'une telle numérotation n'est **pas nécessaire** pour obtenir un algorithme dans **L**, autrement dit, pourquoi le problème original CYCLE NON ORIENTÉ est lui-même dans **L**.
5. On considère maintenant la version de ce problème dans un graphe orienté. Dans un tel graphe, un cycle orienté est une suite finie d'arêtes e_1, \dots, e_k deux à deux distinctes, consécutives, telles que $e_i = s_i \rightarrow s_{i+1}$, avec $s_{k+1} = s_1$ (le cycle revient à son point de départ, comme en version non orientée). Donnez la complexité du problème suivant, en justifiant :

Entrée : Un graphe *orienté* G .

Question : Existe-t-il un cycle dans G ?