

TD 1 : Problèmes de décision et machines de Turing

Théorie de la complexité S5.

2022

Exercice 1 — *Décrire un problème de décision*

Décrivez les problèmes suivants formellement sous forme d'un problème de décision. Précisez l'ensemble \mathcal{L} des instances et la question à laquelle il faut répondre.

1. (BIP ?) : Savoir si un graphe est biparti.
2. (SAT ?) : Connaissant une formule booléenne déterminer comment la rendre vraie.
3. (3SAT ?) : SAT restreint aux formes normales conjonctives où les clauses ont 3 variables.
4. (ADD ?) : Addition de deux entiers.
5. (TAU ?) : Connaissant une formule booléenne déterminer si elle est toujours vraie.
6. (3-TAU ?) : TAU restreint aux formes normales disjonctives où les clauses ont 3 variables.
7. (QBF ?) : Déterminer si une formule booléenne quantifiée est vraie.
8. (CONNECTIVITY ?) : Connexité d'un graphe.
9. (PLANARITY ?) : Planarité d'un graphe.
10. (k-COL ?) : k-colorabilité d'un graphe.
11. (CHROMA) : Nombre chromatique d'un graphe.
12. (SIZE) : Nombre de nœuds d'un graphe.
13. (INS) : Nombre de stabilité d'un graphe.
14. (WINS) : Trouver un stable de poids maximum dans un graphe.
15. (MATCHING) : trouver un couplage maximum dans un graph.
16. (VERTEX COVER) : trouver le nombre minimum de nœuds d'un graphe nécessaires pour couvrir toutes les arêtes.
17. (HAM ?) : existence d'un chemin hamiltonien.
18. (COHAM ?) : déterminer si un graphe n'est pas hamiltonien.
19. (SHP) : Trouver un plus court chemin élémentaire entre deux nœuds, poids positifs.
20. (LOP) : Trouver un plus long chemin élémentaire entre deux nœuds, poids positifs.
21. (TSP) : Trouver un cycle de poids minimum passant par chaque nœud d'un graphe non orienté complet pondéré.
22. (MSPT) : Trouver un arbre couvrant de poids minimum d'un graphe.
23. (SET COVER) : Trouver une couverture minimum d'un ensemble X à l'aide des ensemble d'une partie de $\mathcal{P}(X)$.
24. (UST) : Trouver un arbre de poids minimum couvrant une partie des nœuds d'un graphe.
25. (DST) : Trouver une arborescence de poids minimum couvrant une partie des nœuds d'un graphe orienté et enracinée en un nœud particulier du graphe.
26. (GRUNDY ?) : Existence fonction de Grundy d'un graphe.
27. (SUBSET SUM ?) : Déterminer s'il est possible d'atteindre une certaine valeur en additionnant une partie des entiers d'un ensemble.
28. (PARTITION ?) : Déterminer s'il est possible de partitionner un ensemble d'entiers en 2 sous-ensembles de même somme.

29. (KNAPSACK) : Problème du sac à dos.
30. (LP) : Résoudre un programme linéaire en nombre réels.
31. (ILP) : Résoudre un programme linéaire en nombre entiers.
32. (FLOW) : Trouver un flot maximum dans un réseau.
33. (MREG ?) : Déterminer si une chaîne de Markov est régulière.
34. (CHESS ?) : Déterminer si une position sur un plateau d'échec est gagnante pour le joueur blanc.
35. (REVERSI ?) : Déterminer si une position sur un plateau de reverso est gagnante pour le joueur blanc.
36. (SUDOKU ?) : Résoudre un sudoku.
37. (U-SUDOKU ?) : Unicité de la solution d'un sudoku.
38. (MIN SUDOKU) : Déterminer combien d'indices minimum sont nécessaires à une configuration de sudoku.

Exercice 2 — *Machines de Turing*

Dans cet exercice, on pourra supposer vraies toutes les équivalences de l'exercice 3.

1. Écrire une machine de Turing déterministe qui remplace l'entier en entrée par des 1
2. Écrire une machine de Turing déterministe qui vérifie si un mot est un palindrome
3. Écrire une machine de Turing déterministe qui résout le problème (ADD ?)
4. Écrire une machine de Turing non déterministe qui résout le problème (SAT ?)
5. Écrire une machine de Turing déterministe qui résout le problème (SAT ?)
6. Écrire une machine de Turing déterministe qui résout le problème (3-COL ?)
7. Écrire une machine de Turing non déterministe qui résout le problème (W-INS)

Exercice 3 — *Équivalences entre machines de Turing*

1. Soit une machine de Turing \mathcal{M} avec un alphabet Σ , simulez \mathcal{M} avec une machine \mathcal{M}' classique.
2. Soit une machine de Turing \mathcal{M} , simulez \mathcal{M} avec une machine \mathcal{M}' qui ne possède que des 1 et des 0.
3. Comment encoder la réponse OUI ou NON d'un problème de décision par une machine de Turing sans état d'acceptation ou de refus ?
4. Soit une machine de Turing \mathcal{M} , simulez \mathcal{M} avec une machine \mathcal{M}' ayant une demi-bande (infinie vers la droite, finie vers la gauche).
5. Soit une machine de Turing \mathcal{M} avec 2 bandes, simulez \mathcal{M} avec une machine \mathcal{M}' classique.
6. Soit une machine de Turing \mathcal{M} avec une bande en deux dimensions, simulez \mathcal{M} avec une machine \mathcal{M}' classique.
7. Soit une machine de Turing \mathcal{M} non déterministe, simulez \mathcal{M} avec une machine \mathcal{M}' déterministe.