

# Evaluation du cours 5

Nom :

Prénom :

Répondez sur cette première page, les énoncés sont sur les pages suivantes.

## **Question 1 (V-5.1) : Algorithme glouton pour le voyageur de commerce**

## **Question 2 (V-5.2) : Précision des algorithmes gloutons**

1. Un algorithme glouton donne toujours un résultat non exact
2. La philosophie d'un algorithme glouton est de prendre des décisions localement optimales
3. Un algorithme glouton est d'autant plus précis qu'il est complexe

## **Question 3 (V-5.3) : Complexité d'un algorithme glouton**

1.  $\Theta(n^2)$
2.  $\Theta(n)$
3.  $\Theta(n!)$

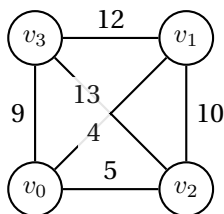
## **Question 4 (B-5.1) : Algorithmes gloutons pour le voyageur de commerce**

## **Question 5 (B-5.2) : Précision des algorithmes gloutons**

## **Question 6 (B-5.3) : Complexité maximale d'un algorithme glouton**

## [Vert] Question 1 : Algorithme glouton pour le voyageur de commerce

Donner la longueur réponse au problème du voyageur de commerce (sans retour au sommet initial) en partant du sommet  $v_0$  donnée par l'algorithme glouton qui choisit systématiquement le sommet le plus proche non visité dans le graphe suivant :



## [Vert] Question 2 : Précision des algorithmes gloutons

Entourer les propositions qui sont vraies.

## [Vert] Question 3 : Complexité d'un algorithme glouton

On considère un algorithme glouton pour le problème du voyageur de commerce sur un graphe à  $n$  sommets. On considère trois heuristiques dont les complexités sont indiquées dans la liste donnée. Entourez celles qui peuvent potentiellement être intéressantes.

## [Bleu] Question 4 : Algorithmes gloutons pour le voyageur de commerce

Estimer (au mieux) la complexité totale d'un algorithme glouton pour la résolution du voyageur de commerce sur un graphe, dont la complexité à chaque étape est en  $\mathcal{O}(n^4)$ ,  $n$  étant le nombre de sommets du graphe.

## [Bleu] Question 5 : Précision des algorithmes gloutons

Prenons un graphe non pondéré, simple, symétrique et complet. Sur le problème de la recherche d'un plus court chemin d'un sommet de départ à un sommet d'arrivée (différent du sommet de départ), on choisit un algorithme glouton avec une très mauvaise heuristique : à chaque étape il choisit un sommet aléatoirement pour se déplacer. Sur un graphe à  $n = 288$  sommets, quelle est l'erreur maximale (écart entre la réponse donnée par l'algorithme glouton et celle donnée par un algorithme sans erreur) faite par cet algorithme?

## [Bleu] Question 6 : Complexité maximale d'un algorithme glouton

On considère un algorithme glouton dont l'objectif est de factoriser un entier donné en entrée. L'idée est de tester tous les entiers en partant de 2 et en incrémentant jusqu'à trouver un diviseur, puis une fois un diviseur trouvé de s'appeler récursivement. Un programme python implémentant cet algorithme est donné ci-dessous :

```
compteur=0
def factor(n):
    for i in range(2,n):
        compteur = compteur + 1
        if n % i == 0:
            print(i)
            if n != i:
                factor(n//i)
```

Dans ce programme, la variable `compteur` compte le nombre d'opérations effectuées. Pour quelle valeur d'entrée comprise entre 1 et 78 l'appel à la fonction `factor` augmentera le plus la variable `compteur` (on ne demande aucune justification)?