

Exercice 1 : Questions de cours (05 points)

- 1) Que satisfait la solution optimale x^* du problème d'optimisation : $\max_{x \in S} f(x)$?
- 2) Citer un exemple d'un problème de la classe P et un exemple d'un problème de la classe NP-Complet.
- 3) Deux algorithmes A1 et A2 de complexités respectives $O(n^* \log n)$ et $O(n + \log n)$. Lequel est plus efficace ? justifier.
- 4) Pour appliquer la méthode Branch and Bound à un problème d'optimisation combinatoire on utilise une borne inférieure et une borne supérieure de sa solution.

Compléter chaque case du tableau ci-dessous par l'une de ces deux bornes :

	Solution non faisable	Solution approchée
Cas de maximisation		
Cas de minimisation		

Exercice 2 : (05 points)

Considérons une instance du PVC symétrique avec n le nombre de villes et $d[][]$ la matrice des distances.

- 1) Comment représenter une solution pour ce problème.
- 2) Rappeler le nombre de solutions candidates.
- 3) Ecrire l'algorithme retournant la distance totale d'une solution x puis évaluer sa complexité.

Exercice 2 : (10 points)

Etant donnés n articles chacun ayant un revenu de transport v_i dinars et un poids w_i kg. On désire déterminer quels articles transporter par un camion de capacité C kg ($C < \sum_{i=0}^{n-1} w_i$) afin de maximiser le revenu total de transport.

- 1) Quel problème vu au cours pouvant être considéré comme modèle pour ce problème ?
- 2) Déduire la classe de complexité de ce problème.
- 3) Ecrire la formulation mathématique de ce problème.
- 4) Calculer la taille de l'espace de recherche.
- 5) L'algorithme ci-contre constitue une partie de la résolution de ce problème.
 - a) Que représente le vecteur binaire x pour le problème donné.
 - b) Que fait cet algorithme ? Evaluer sa complexité.

```
boolean ALGO (int n , int [] w , binary [] x , int C)
{ s = 0 ;
  for ( i = 0 ; i < n ; i++)
    s += x[i] * w[i] ;
  return (s <= C) ; }
```

- 6) Ecrire l'algorithme qui calcule le revenu total de transport pour une solution x puis évaluer sa complexité.
- 7) Appliquer la méthode de la programmation dynamique à l'instance suivante :

Article	1	2	3	4
v_i	8	6	10	5
w_i	2	3	1	4
$C = 7$				