Université MB de M'sila
Faculté de MI
Département d'informatique
Formation Master 1 IDO

Module Méthodes d'optimisation
Examen session normale janvier 2024
Durée : 1 heure 30 minutes

Exercice 1 (questions de cours / 04 points)

1) $\forall x \in S : f(x^*) \leq f(x)$ 1
2) **La solution d'un sous-problème ne dépend que de celle de prédécesseur (hypothèse de Markov)** 1
3) **Non. KSP. CC** 1
4) **Segmentation de l'ensemble des solutions (Branch) et Heuristiques pour calculer les LB et UB (Bound) et.** 1

Exercice 2 (07 points)

1) **R=(2,1,0,3)** 1
2)
```
Int[] Algo(int k, int M, int[] C) {
s=0;
for (i = 0; i < n; i++) {
      while (M >= C[i]) {
         M -= C[i];
         R[i]++;   }
      s+= R[i];  }
   return R; s }
```
..... 1
3) **O(n.M).** 0.5
4)
   a) **Rendu de monnaie** 1
   b) **Glouton** 1
   c) **Borne supérieure puisqu'il s'agit d'un problème de minimisation.** 1
   d) **Borne inferieure = plafond (M/C[0]) =** $\left\lceil \frac{M}{C[0]} \right\rceil$ 1.5

Exercice 3 (9 points)

1) **Espace de recherche =** $\{0,1\}^n$**, sa taille =** 2^n**, la variable de décision =** $x = (x_i); i = 1, n$.
   Fonction objectif = $f(x) = \sum_{i=1}^{n} x_i a_i$**, les contraintes =** $f(x) = \sum_{i=1}^{n} x_i a_i$ et $x_i \in \{0,1\}$
   Type de P = combinatoire 3.5
2) $x0 = 1001$ et $f(x0) = 7$ 1
3)
   a)
```
Int test(int[] x) {
s=0;
for (i = 0; i < n; i++)
      s+= x[i]* a[i];
if (s > B) return false;
else return true; }
```
```
Int test(int[] x) {
s=0;
for (i = 0; i < n; i++)
      {s+= x[i]* a[i];
       if (s > B) return false;}
return true; }
```
..... 1.5
   b)
```
Int f(int[] x) {
s=0;
for (i = 0; i < n; i++)
      s+= x[i]* a[i];
return s; }
```
..... 1.5
   c) **DP[i][j] = 0 si i=0 ou j= 0**
   Pour (i= 1,n ; j= 1, B) : DP[i][j] = DP[i-1][j] si $a_i > j$
   DP[i][j] = max (DP[i-1][j] ; DP[i-1][j-a_i**]+** a_i**) si** $a_i \leq j$ 1.5