**Examen continu en ROC- M1 STIC-2020**

**Questions de cours ( 4 pts)**

1. Expliquer les étapes de résolution d’un problème d’optimisation.
2. Quelle est la différence entre l’optimisation continue et l’optimisation combinatoire ?
3. Expliquer comment optimiser une fonction multi-variable sans contraintes.

**Exercice 1  ( 4 pts ):**

Au ministère de l'agriculture, on a établi la fonction de profit suivante pour les fermes cultivant des graines de soja et des pistaches :



où *P*(*x*, *y*) sont les profits annuels en DA, *x* représente le nombre d'acres plantés en grains de soja, et *y* donne le nombre d'acres plantés en pistaches. Le fermier possède une terre de 500 acres. Pour des raisons de stockage, la superficie plantée en pistaches doit être deux fois plus grande que celle plantée en soja, donner la modélisation du problème et expliquer sans calcul comment trouver la solution dans ce cas.

**Exercice2 ( 6 pts):**

Le problème de bin packing consiste à ranger un ensemble d’objets O1, O2,…Oi de poids différents W1, W2,….Wi respectivement dans des Bins ( Boites) avec une capacité fixe C avec . L’objectif est de trouver le nombre minimum de Bins à utiliser pour ranger tous les objets.

Soit l’exemple suivant, on a 6 objets avec les poids respectivement wieght[] = {4, 8, 1, 4, 2,1} et la capacité du Bin est C = 10.

**Questions :**

1. Ecrire un modèle linéaire en nombres entiers modélisant le problème du bin packing.
2. Appliquer l’heuristique *first fit* (ranger l’objet dans le premier bin qui peut le contenir) pour résoudre l’exemple précédant, et donner le nombre de bins utilisés.
3. Si une borne inferieure LB pour ce problème peut être calculée comme suit :

Expliquer comment peut-on résoudre exactement ce problème en utilisant la méthode Branch and Bound.

**Exercice 3 (6 pts):**

Le problème de la plus longue sous-séquence commune à deux séquences de caractères (LCS Longest Common Subsequence ) peut être défini comme suit :

Étant donné deux séquences, le problème LCS consiste à trouver la longueur de la sous-séquence la plus longue présente dans les deux séquences. Une sous-séquence est une séquence qui apparaît dans le même ordre relatif, mais pas nécessairement contiguë. Par exemple, "abc", "abg", "bdf", "aeg", " acefg ", .. etc sont des sous-séquences de" abcdefg ".,

**Exemples de la plus longue sous-séquence commune :**

LCS pour les séquences d'entrée «AGGTAB» et «GXTXAYB» est «GTAB» de longueur 4.

Ce problème peut être résolu par la programmation dynamique, pour cela la sous-structure optimale peut être définie comme suit :

Soit respectivement les séquences d'entrée X et Y de longueurs m et n respectivement. Et soit L (X ,Y ) la longueur de LCS des deux séquences X et Y. Voici la définition récursive calculant L (X [1 .. m], Y [1..n]) :

* Si les derniers caractères des deux séquences correspondent (X [m] == Y [n]), alors :

L (X [1..m], Y [1..n]) = 1 + L (X [1..m-1], Y [1..n-1])

* Si les derniers caractères des deux séquences ne correspondent pas (X [m]! = Y [n]), alors :

L (X [1..m], Y [1..n]) = MAX (L(X[1..m-1], Y[1..n]), …………….)

1. Compléter la formule récursive.
2. Écrire un algorithme tabulaire qui résout ce problème.
3. Appliquer cet algorithme sur le l’exemple suivant :   
   X= “AGGTAB” et Y=“GXTXAYB”

**Bon courage**