

TP 1 : Algorithme RT pour l'ordonnancement des tâches

Objectif

- Six tâches doivent être effectuées dans une séquence telle qu'une fonction objective soit maximisée. la fonction objectif peut tirer profit d'un retard réduit.

- Représentation d'une solution pour 6 tâches:

5	3	4	6	1	2
---	---	---	---	---	---

- Structure de voisinage : échange des tâches.

5	3	2	6	1	4
---	---	---	---	---	---

- Une solution a 15 voisins (chacun des 15 échanges possibles des 6 tâches).

Algorithme Recherche Tabou: Mémoire basée sur la **récence** et **classification** tabou.

- Les 3 échanges des tâches les plus récents sont classés comme tabous pour le problème considéré.
- L'occupation tabou sera pour 3 itérations.
- Les critères d'aspiration sont utilisés pour éliminer certains échanges des tâches de tabou.

5	3	4	6	1	2	} with a (4,2) swap, we have (2,4) as tabu for 3 iteration	1 2 3 4 5 6
Obj Fn value = 20							
5	3	2	6	1	4		

Fig. Tabu Structure

(5,4) 5	(3,1) 3	(1,3) -3	(1,3) 5	(5,1) 0
(3,6) 3	(3,2) 1	(4,2) -4	(2,5) 1	(3,6) -1
(3,2) 1	(5,1) -1	(4,5) -6	(1,4) -1	(1,4) -2
(4,1) -4	(6,2) -5	(6,2) -7	(6,4) -3	(3,1) -5
Iteration 0	Iteration 1	Iteration 2	Iteration 3	Iteration 4

Fig. Top 4 candidates in each iteration

Travail à faire :

En reprenant le principe de l'algorithme Recherche Tabou, développer un programme qui permet de donner :

1. Le meilleurs choix à partir de Top 4 candidates pour chaque itération?
2. Leur évaluation **Obj Fn value**?
3. Donnez les états de **la liste Tabou** pour chaque itération?

4. Exécuter le programme pour n itérations avec une solution de taille m (en proposant une fonction d'adaptation convenable au problème considéré) ?
5. Donner le résultat de la Recherche Tabou en numéro d'itération et de valeur d'adaptation à chaque itération?

Bon courage