

TD 8 – Récursion (et comment l'éliminer)

Concepts informatiques (CI2)

2011-2012

1 Coupe en deux

Soit la fonction f définie par \mathcal{F}

On s'intéresse au problème consistant à chercher si il existe un chemin dans un graphe entre deux nœuds i et j donnés. On peut utiliser l'algorithme dit de « parcours en profondeur » :

- on se donne un tableau auxiliaire `connu` de booléens, dont toutes les cases sont initialement à `false`, et qui servira à se souvenir de l'ensemble des sommets que l'on a rencontrés
- on part du nœud i
- on met `connu[i]` à `true`
- on examine chacune des flèches sortant de i : pour chacune d'elle, en notant k son extrémité, on regarde `connu[k]`. Si cette case contient `true`, on ne fait rien. Sinon, on fait un appel récursif sur k .

Ce faisant, on explore tous les sommets du graphe que l'on peut atteindre à partir de i . Si l'on rencontre j au passage, alors on a gagné.

Écrire une méthode récursive implémentant cet algorithme.

Écrire une méthode itérative qui fait la même chose. Pour ce faire, il faut maintenir dans une pile le chemin que l'on a parcouru entre le sommet de départ et le sommet courant.