Diviser pour règner

N. TSOPZE

Département d'Informatique - Université de Yaoundé I

Plan du chapitre

définition

toute fonction ou procédure qui s'appelle elle même.

Exemples:

1. définition des entiers
2. fonction factorielle $n!$
3. tour d'Hanoi
4. ...

Pile

pile d'exécution

emplacement mémoire destinée à mémoriser les paramètres, les variables locales ainsi que les adresses de retour des fonctions en cours d'exécution

- principe LIFO (Last-In-First-Out) : dernier entré premier sorti.
- taille fixe, attention au débordement de pile.

principe

Deux parties

1. Clause de base (test d'arrêt): permet de mettre fin aux appels recursifs et à l'algorithme
2. appel récursif: appel de l'algorithme dans une instruction pour le traitement des autres cas

schéma général

Si si test d'arrêt

instructions du point d'arrêt

Sinon

instructions

Appel récirsif (paramètres changés);

instructions

Plan du chapitre

Principe

étape 1: Diviser le problème global (de grande taille) en sous - problèmes (de faible taille)

étape 2: division du problème en problèmes de taille faible

étape 3: Resoudre les sous - problèmes (soluble de manière triviale ou avec un algorithme simple)

étape 4: Fusionner les solutions des sous problèmes pour former la solution du problème global

Schéma - principe recursif

Clause de base: si la taille des données est petite

solution triviale

division: décomposition du problème, avec éventuellement des instructions

Appel recursif: Appel de l'algorithme pour resoudre les sous - problèmes

Fusion: écrire une procédure qui permet de mettre ensemble les solutions des sous problèmes pour construire l solution globale.

Plan du chapitre

recherche dichotomique

Données	Tableau trié T + valeur v
problème	v est-il dans le tableau T?
Approche	reduire progressivement la taille de tableau jusqu'au tableau singleton, tester le singleton puis reconstituer la solution globale

Expliquer la démarche DPR pour la recherche dans un arbre binaire, dans un arbre binaire de recherche

tri fusion

Données Tableau T de n valeurs en ordre quelconque

problème ranger les éléments de T tel que
$T_i \leq T_{i+1} \forall i 1 \leq i \leq n$

Approche diviser le tableau jusqu'au tableau singleton,
fusionner les tableaux obtenus en conservant le tri

multiplication matricielle

Etant donné deux matrices carrées A et B

1. Ecrire un algorithme qui calcule $C = A * B$
2. calculer la complexité de l'algorithme précédent

multiplication matricielle - Strassen

$$\begin{pmatrix} r & s \\ t & u \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} * \begin{pmatrix} e & f \\ g & h \end{pmatrix}$$

On pose:

$$\begin{cases} p_1 = a(g-h) \\ p_2 = (a+b)h \\ p_3 = (c+d)e \\ p_4 = d(f-e) \\ p_5 = (a+d)(e+h) \\ p_6 = (b-d)(f+h) \\ p_7 = (a-c)(e+g) \end{cases}$$

Calculer r, s, t et u en fonction des p_i puis déduire un algorithme DPR pour calculer le produit matriciel