IUP Miage
Système d'exploitation

TP — Communication de processus

Ce TP illustre l'utilisation des primitives `fork ()`, `wait ()`, et `pipe ()`.

1 Examen des tubes

Il s'agit de mettre en évidence le possible comportement bloquant en écriture lors de l'utilisation d'un tube.

En particulier, écrivez un programme qui détermine la taille de la mémoire associée aux tubes sur le système.

2 Calcul matriciel élémentaire

Le but est de réaliser un programme effectuant une addition de matrices carrées. Le programme sera développé selon le modèle maître-esclave :

- le processus maître est chargé de :
  - lire les deux matrices ;
  - créer les processus esclaves ;
  - envoyer à chaque esclave le travail qu'il doit effectuer ;
  - récupérer les résultats et afficher le résultat final.
- chaque esclave doit effectuer l'addition entre deux lignes des matrices de départ :
  - il récupère les deux lignes de matrices que lui envoie le maître ;
  - il effectue l'addition ;
  - il envoie la ligne résultat au maître.

Le maître et ses esclaves communiquent par tubes UNIX.

Une implantation de ce schéma général doit choisir :

- le nombre de processus de calcul. Il peut être fixe et indépendant du nombre de lignes des matrices, ou fonction de la taille des matrices.
- le protocole d'émission/réception des lignes entre le processus maître et les processus de calcul. Quelles sont les données transmises ? La taille des lignes est-elle transmise ? Les lignes sont-elles transmises l'une à la suite de l'autre ou entrelacées ?
- le maître se charge-t-il de collecter les résultats ou un autre processus en est-il chargé ?
- la terminaison du programme. Les processus connaissent-ils *a priori* le nombre de valeurs (lignes) à traiter ou un autre mécanisme de terminaison est-il utilisé ?

Id: tppipe.tex,v 1.6 2002/05/24 07:24:34 marquet Exp