

Documentation pour le mini-projet Erlang

NASSABAIN Leonardo M1 SDSC

Pour ce projet, il fallait implémenter les estampilles matricielles en Erlang, afin de synchroniser la communication d'un ensemble de processus. Au départ, j'envisageais de représenter les matrices sous forme de listes à deux dimensions, c-à-d sous forme de liste de listes. Mais cela rendait leur manipulation compliquée et fastidieuse. J'ai donc décidé d'implémenter les matrices sous forme de listes à une dimension, en les "aplatissant". Par exemple, une matrice 3 x 3 devient une liste de taille 9.

Remarques:

Dans le programme, chaque processus envoie un message à chacun des autres processus. Pour lancer le programme, il suffit de compiler le fichier **erl**, puis d'appeler la fonction **test** du module avec le nombre de processus voulu.

Par exemple, si on veut lancer 3 processus il faut faire:

```
>c(projet).  
>projet:test(3).
```

Cependant, quelques heures avant de rendre le projet, je me suis rendu compte qu'il y parfois des problèmes de concurrence, il me semble que certains processus se terminent et qu'après, il n'est plus possible de communiquer avec eux par leur nom. Mais malheureusement, je n'ai pas eu le temps de trouver la solution au problème.

make_list(N)

La fonction va créer de manière récursive une liste de taille N, chaque élément sera initialisé à 0 et cette liste sera renvoyée.

Paramètres:

N, la taille de la liste que l'on veut créer.

unflatten_matrix(Matrix, Size)

La fonction va découper la liste à une dimension Matrix, en sous-listes de taille Size et va renvoyer une liste à 2 dimensions (une matrice).

Paramètres:

Matrix, liste à 1 dimension

Size, dimension de la matrice carrée qui sera renvoyée

find_index(Index1, Index2, Size)

La fonction renvoie l'indice d'une position dans une matrice si on l'aplatissait.

Par exemple si on avait la matrice suivante: $\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$, "f" se trouve à la position (2,3). Cette

$\begin{bmatrix} d & e & f \\ g & h & i \end{bmatrix}$

Fonction nous renverrait 6 car 6 est l'indice de 4 dans la matrice aplatie: [a,b,c,d,e,f,g,h,i].

Paramètres:

Index1 l'indice de la ligne

Index2 l'indice de la colonne

Size la dimension de la matrice carrée

incr_mat_send(Matrix, Index1, Index2, Size)

La fonction effectue les incréments nécessaires avant l'envoi d'un message vers un processus donné.

Paramètres:

Matrix, la matrice à modifier

Index1, le numéro du processus émetteur

Index2, le numéro du processus récepteur

Size, la dimension de la matrice carrée Matrix

incr_mat_rec(Matrix, Index1, Index2, Size)

La fonction effectue les incréments nécessaires après la réception d'un message par un processus.

Paramètres:

Matrix, la matrice à modifier

Index1, le numéro du processus récepteur

Index2, le numéro du processus émetteur,

Size, la dimension de la matrice carrée Matrix

max_matrice(Mat1, Mat2, IndexI, IndexJ, Size)

La fonction va comparer les éléments des matrices Mat1 et Mat2 deux par deux et renvoyer une matrice contenant le maximum pour chaque indice. Cependant, les éléments se trouvant aux positions (IndexI, IndexI) et (IndexJ, IndexI) ne seront pas comparés, les éléments de la matrice Mat1 seront pris.

Paramètres:

Mat1, la première matrice

Mat2, la deuxième matrice

IndexI et IndexJ, indices utilisés pour retrouver les positions à ne pas comparer,

Size, la dimension des matrices carrées Mat1 et Mat2

spawnN(N)

Fonction utilisée pour créer N processus, et leur donner un nom.

Paramètres:

N, le nombre de processus à créer

fill_list(N)

Fonction utilisée pour créer une liste contenant tous les éléments allant de N à 1.

Paramètres:

N, le nombre qui définit la taille de la liste et la borne supérieure des éléments qui seront présents

cond1_rec(Mat1, Mat2, IndexI, IndexJ, N)

Fonction utilisée pour vérifier la première condition nécessaire à la réception d'un message

Paramètres:

Mat1, l'estampille du message reçu

Mat2, l'horloge matricielle du processus récepteur

IndexI, IndexJ, indices nécessaires pour vérifier la première condition de réception

N, la taille des matrices carrées Mat1 et Mat2

cond2_rec(Mat1, Mat2, IndexI, IndexJ, N)

Fonction utilisée pour vérifier la deuxième condition nécessaire à la réception d'un message

Paramètres:

Mat1, l'estampille du message reçu

Mat2, l'horloge matricielle du processus récepteur

IndexI, IndexJ, indices nécessaires pour vérifier la deuxième condition de réception

N, la taille des matrices carrées Mat1 et Mat2

sendMsg(ListDest, Matrix, Id, N)

L'une des fonctions principales du programme. Utilisée par un processus pour envoyer des messages au processus dont les numéros se trouvent dans la liste ListDest.

Paramètres:

ListDest, liste contenant les numéros de processus vers lesquels le processus émettra de message

Matrix, l'horloge matricielle du processus courant

Id, le numéro du processus courant

N, la dimension de la matrice carrée Matrix

receiveMsg(Matrix, Id, N, Cpt)

Une des fonctions principales du programme. La fonction est utilisée pour recevoir un nombre Cpt de messages. Dans la fonction, les conditions de réception de message sont vérifiées avant la livraison. Si les conditions sont satisfaites, il y a livraison, sinon un message de type "waiting" sera envoyé tant que les conditions ne sont pas bonnes. Une fois qu'elle sont satisfaites, un message normal est envoyé avec les mêmes contenus.

Paramètres:

Matrix, l'horloge matricielle du processus courant (récepteur)

Id, le numéro du processus courant,

N, la dimension de la matrice carrée Matrix,

Cpt, le nombre de messages à recevoir

process(Id, Matrix, N)

La fonction qu'appelle chaque processus à sa création et qui gère l'envoi et la réception des différents messages.

Paramètres:

Id, numéro du processus

Matrix, l'horloge matricielle du processus

N, la dimension de la matrice carrée Matrix

test(N)

La fonction exportée, qu'on appelle afin de lancer le programme.

Paramètres:

N, le nombre de processus à lancer