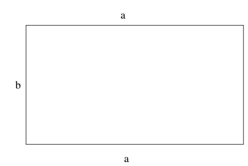
Université Bordeaux I, ENSEIRB-MATMECA Année 2010
U.F.R de Mathématiques et Informatique

PROJET de Programmation CHP

MIM42
Calcul Haute Performance

On se place dans le domaine $[0, L_x] \times [0, L_y]$ de $\mathbb{R}^2$ dans lequel on résoud l'équation de la chaleur:



$$\partial_t u(x, y, t) - D\Delta u(x, y, t) = f(x, y, t) \tag{1}$$
$$u|_{\Gamma_0} = g(x, y, t) \tag{2}$$
$$u|_{\Gamma_1} = h(x, y, t) \tag{3}$$

1. **Analyse du problème**
   On se propose de résoudre numériquement cette équation par la méthode des différences finies.

   (a) Ecrire le schéma d'Euler implicite à l'aide de différences finies centrées du second orde en espace.

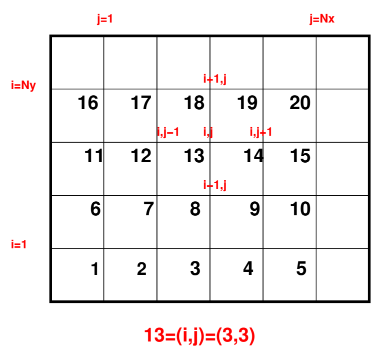


Figure 2: Principe de la numérotation globale et locale

   (b) En utilisant la numérotation proposée figure 2, montrer que le schéma précédent se met sous la forme matricielle $AU = F$.

   (c) Décrire précisément la structure de la matrice A.

   (d) Enoncer les propriétés de la matrice A.

2. **Implémentation informatique**
   On utilisera les cas test suivants pour valider le travail:
   $L_x = L_y = 1.0$, $D = 1.0$

La solution stationnaire résultant des conditions suivantes

$$f = 2(y - y^2 + x - x^2) \qquad g = 0 \qquad h = 0 \tag{4}$$

Puis

$$f = sin(x) + cos(y) \qquad g = sin(x) + cos(y) \qquad h = sin(x) + cos(y) \tag{5}$$

La solution instationnaire périodique résultant des conditions suivantes

$$f = e^{-\left(x - \frac{Lx}{2}\right)^2} e^{-\left(y - \frac{Ly}{2}\right)^2} cos(\frac{\pi}{2}t) \qquad g = 0 \qquad h = 1 \tag{6}$$

(a) **Ecrire le code séquentiel**

i. Cahier des charges pour le nom des variables: Nx nombre de noeuds dans la direction x, Ny nombre de noeuds dans la direction y, A matrice, U solution au temps n, Uo solution au temps n-1, dx,dy,dt pas d'espace et de temps, Lx, Ly, D

ii. Nx,Ny,Lx,Ly,D seront lues dans un fichier de paramètres.

iii. Construire un module contenant les trois fonctions: f, g et h.

iv. Construire un module contenant le gradient conjugué.

v. Construire une subroutine pour remplir la matice A (on pourra coder la fonction ou subroutine permettant de passer de la numérotation globale à la numérotation locale (i,j) voir figure 2).

(b) **Ecrire le code parallèle**

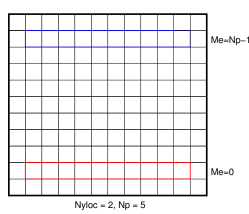


Figure 3: Exemple de répartition des inconnues sur les processeurs

i. On s'arrangera pour que Ny=Np*Nyloc (voir figure 3), Nyloc sera lue à la place de Ny.

ii. Chaque processeur remplira A(i1:iN,Ns) avec Ns le nombre global d'inconnues du problème.

iii. L'algorithme du gradient conjugué devra être parallélisé (Algorithmique parallèle: quelles sont les opérations nécessitant des communications?).

iv. Chaque processeur écrira sa partie de solution dans son fichier sol00Me.dat (on pourra utiliser la subroutine Rename proposée en annexe).

Il faudra fournir les documents suivants:

1. Un rapport contenant

- L'analyse mathématique du problème.

- Une description détaillée des communications réalisées.
- Le processus de validation du code.
- Les courbes du temps de calcul en fonction du nombre de processeurs (Speed-up, Efficacité, ...).
- Vos conclusions.

2. Le code // documenté et commenté, avec les commandes de compilation.

Annexe:

```
subroutine Rename(Me,name)
   implicit none
   integer ::  Me
   character*13 ::name
   character*3 ::  tn
   integer ::  i1,i2,i3
   i1 = Me/100
   i2 =( Me - 100*i1)/10
   i3 = Me - 100*i1 -10*i2
   tn = char(i1+48)//char(i2+48)//char(i3+48)
   name='sol'//tn//'.dat'
end subroutine Rename
```