	<p align="center">ING2 : EXAMEN DE RATTRAPAGE PROGRAMMATION PARALLELE</p> <p align="center">EXAMEN PAPIER (AUCUN DOCUMENT N'EST AUTORISE)</p> <p align="center">DUREE 2 HEURES</p>	
Rédigé par : équipe pédagogique du cours de programmation parallèle.	Ref : <i>ING2-PROGPARA</i>	
A l'intention de : Etudiants d'ING2 MI	Créé le : 12/05/2017	

Exercice 1 (3Pts)


Un élève a donné sa solution à un exercice dont le sujet est le suivant : Ecrire un programme qui déclare 3 tableaux a, b et c puis initialise le contenu des 2 premiers tableaux, calcule le contenu du troisième $c = a + b$ et affiche à la volée le contenu du vecteur c.

```
#include <omp.h>
#include <stdio.h>
#include <stdlib.h>
#define N    50
#define CHUNKSIZE  5

int main (int argc, char *argv[])
{
    int i, chunk, tid;
    float a[N], b[N], c[N];
    /* Some initializations */
    for (i=0; i < N; i++)
        a[i] = b[i] = i * 1.0;
    chunk = CHUNKSIZE;

    #pragma omp parallel for \
        shared (a,b,c,chunk) \
        private (i,tid) \
        schedule(static, chunk)
    {
        tid = omp_get_thread_num();
        for (i=0; i < N; i++)
        {
            c[i] = a[i] + b[i];
            printf("tid= %d i= %d c[i]= %f\n", tid, i, c[i]);
        }
    }
}
```

Question : Ce code donne-t-il le résultat voulu ? Sinon corrigez le ?

	ING2 : EXAMEN DE RATTRAPAGE PROGRAMMATION PARALLELE EXAMEN PAPIER (AUCUN DOCUMENT N'EST AUTORISE) DUREE 2 HEURES
Rédigé par : équipe pédagogique du cours de programmation parallèle.	Ref : <i>ING2-PROGPARA</i>
A l'intention de : Etudiants d'ING2 MI	Créé le : 12/05/2017

Exercice 2 (3pts)

Ecrire un programme qui calcule la racine carrée de la somme des carrés d'un vecteur V de longueur N connue. L'initialisation du vecteur V se fait par le thread principal.

$$\text{sommeCarre} = (\text{Sum}(x_i * x_i))^{1/2}$$

- a- sur une communication point à point classique
- b- sur une communication point à point sur une structure d'anneau

Exercice 3 : (8pts) la résolution numérique de l'équation de Laplace (ou équation de la chaleur).

Imaginons-nous une plaque métallique illimitée à laquelle on applique une source de chaleur sur une section d'un bord. La chaleur va se propager le long de la barre en suivant une dynamique décrite par l'équation différentielle partielle suivante :

$$\partial_t \rho(x,t) = \nabla^2 \rho(x,t)$$

où "ρ" est la température. Si la température aux bords est maintenue constante, la distribution de chaleur dans la plaque tend vers un état stationnaire. Dans ce cas, la dérivée temporelle dans l'équation précédente disparaît, et on se retrouve avec l'équation de Laplace :


$$\nabla^2 \rho(x,t) = 0$$

On se propose de résoudre l'équation de Laplace en simulant l'équation de la chaleur. En commençant par une distribution de chaleur identiquement nulle, on applique une température constante aux 2 bords et on fait évoluer l'équation de la chaleur jusqu'à ce que la dynamique se stabilise. Voici un schéma numérique qui permet de résoudre l'équation de la chaleur itérativement :

$$\rho^{t+1}(i, j) = (\rho^t(i+1, j) + \rho^t(i-1, j) + \rho^t(i, j-1) + \rho^t(i, j+1) + 4 * \rho^t(i, j)) / 8$$

Les indices "i" et "j" sert à numérotter les points de l'espace discret.

1	1	1	1	1	1
1	0	0	0	0	0
1	0	0	0	0	0
1	0	0	0	0	0
1	0	0	0	0	0
0	0	0	0	0	0

 <p>EISTI Ecole Internationale des Sciences du Traitement de l'Information</p>	<p align="center">ING2 : EXAMEN DE RATTRAPAGE PROGRAMMATION PARALLELE</p> <p align="center">EXAMEN PAPIER (AUCUN DOCUMENT N'EST AUTORISE)</p> <p align="center">DUREE 2 HEURES</p>	
Rédigé par : équipe pédagogique du cours de programmation parallèle.	Ref : <i>ING2-PROGPARA</i>	
A l'intention de : Etudiants d'ING2 MI	Créé le : 12/05/2017	

Questions

1. Ecrivez un programme en C ou en C++ qui résout le problème de l'équation de la chaleur à 2 dimensions, on choisit une discrétisation de N sur l'axe x et y et une discrétisation temporelle de T. (2pts)
2. OpenMP
 1. Parallélisez le programme par des directives OpenMP (3pts)
3. MPI
 1. Donnez le code de cette parallélisation avec MPI. (3pts)

Exercice 4 : communication collective (6 pts)

Ecrire un programme qui déclare une matrice M de tailles connues NxN et un vecteur V de taille N, et initialise aléatoirement la matrice M par des entiers compris entre 0 et 100. Utilisez les primitives de communication collective pour remplir le vecteur V de la manière ci-dessous.

