Noms membres del grup: Ona Sánchez Núñez, 1601181

**Exercicis Pràctics MPI I**

L’objectiu principal d’aquests exercicis és que experimenteu amb les capacitats d’MPI fent servir casos simples, facilitant d’aquesta manera la transició entre els continguts discutits en les classes de teoria i problemes i la seva aplicació al cas pràctic (més complex) treballat en el laboratori.

En segon lloc, volem que visualitzeu algunes de les propietats i característiques d’MPI que poden estar més relacionades amb el cas pràctic que heu de resoldre al laboratori.

El plantejament dels exercicis i la mecànica de treball per resoldre’ls consisteixen en:

1. Per cada apartat, es proporciona un problema que haureu de paral·lelitzar fent servir MPI, alguns amb comunicació punt a punt i un altre amb comunicació col·lectiva.
2. Per cada apartat, es el fragment de codi que resol el problema.
3. La solució als exercicis inclourà aquest document que s’haurà de lliurar com a .pdf i els codis que genereu pels diferents apartats.

**Exercicis**

1. **Comunicació punt a punt**

**Escrigui un programa que implementi un ping-pong entre dos processos. El ping-pong consistirà en que el procés amb rang 0 enviï un missatge de 128 bytes al procés amb rang 1 (el contingut del missatge és irrellevant) i que, un cop rebut, el procés 1 torni el mateix missatge al procés amb rang 0. El programa rebrà com a argument un enter que indica quants cops es repeteix el ping-pong. El procés amb rang 0 mesurarà el temps que triga cada interacció (temps d’anada i tornada del missatge) i, un cop fetes totes les interaccions (nombre de repeticions indicat), imprimirà per pantalla la mitjana del temps per interacció.**

* + 1. Quin és el codi del vostre programa? (adjunteu-lo al lliurament)
    2. Feu vàries proves amb diferents nombres . Els resultats obtinguts són consistents? Justifiqueu la vostra resposta i indiqueu si heu fet algun tipus de canvi en el codi inicial.

Cops que es repeteix el ping-pong 🡪 temps mitjà de cada rang

10 🡪 0.000006 i 0.000009

100 🡪 0.000002 i 0.000002

1000 🡪0.000001 i 0.000001

1000000 🡪 0.000001 i 0.000001

Observem que com més iteracions es fa del ping-pong, menys temps es triga de mitja en cada iteració, això es degut a que cada cop la comunicació triga menys, és a dir, les primeres iteracions tindran un temps mitjà més elevat que les últimes, pel que en fer la mitja del temps disminuirà.

* + 1. Quin és el temps mitjà per interacció en el cluster aolin si els dos processos són en el mateix node?

-

* + 1. Quin és el temps mitjà per interacció en el cluster aolin si els dos processos són en nodes diferents?

-

**NOTA: per saber en quin node s’executa un procés podeu fer servir la funció MPI\_Get\_processor\_name.**

1. **Comunicació punt a punt i col·lectiva**

**Implementeu una funció que, a partir d’un vector de dimensió n, distribuït entre p processos de forma cíclica per blocs, faci las comunicacions necessàries per tal que tots els processos acabin amb una còpia del vector complert.**

**La capçalera de la funció serà:**

**void comunica\_vector(double vloc[], int n, int b, int p, double w[])**

**/\* vloc: part local del vector v inicial**

**n: dimensió global del vector v**

**b: mida de bloc que es va fer servir en la distribució del vector v**

**p: número de processos**

**w: vector de longitud n, on s’emmagatzemarà la còpia del vector v complert**

**\*/**

**Teniu el codi del programa en l’arxiu cíclic-mpi.c**

* + 1. Proposeu una solució amb comunicacions punt a punt: Quin és codi del vostre programa? (adjunteu-lo al lliurament)
    2. Proposeu una solució amb comunicacions col·lectives: Quin és codi del vostre programa? (adjunteu-lo al lliurament)

1. **Comunicació col·lectiva**

**El següent fragment de codi calcula la norma de Frobenius de una matriu quadrada.**

**int i, j;**

**double s, norm, A[N][N];**

**Init\_Mat(A);**

**s = 0.0;**

**for (i=0;i<N;i++)**

**for (j=0;j<N;j++) s += A[i][j]\*A[i][j];**

**norm = sqrt(s);**

**printf("norm=%f\n",norm);**

**Implementeu un programa paral·lel en MPI que calculi la norma de Frobenius, de manera que el procés amb rang 0 inicialitzi la matriu A, la reparteixi distribuint les files de la matriu entre tots els processos (ell inclòs), y ﬁnalment obtingui el**

**resultat s i el mostri per pantalla. Només fent servir comunicacions col·lectives. Assumiu que el nombre de files es divisible entre el nombre de processos.**

* + 1. Quin és el codi del vostre programa? (adjunteu-lo al lliurament)