

Nom i Cognoms: \_\_\_\_\_

DNI: \_\_\_\_\_

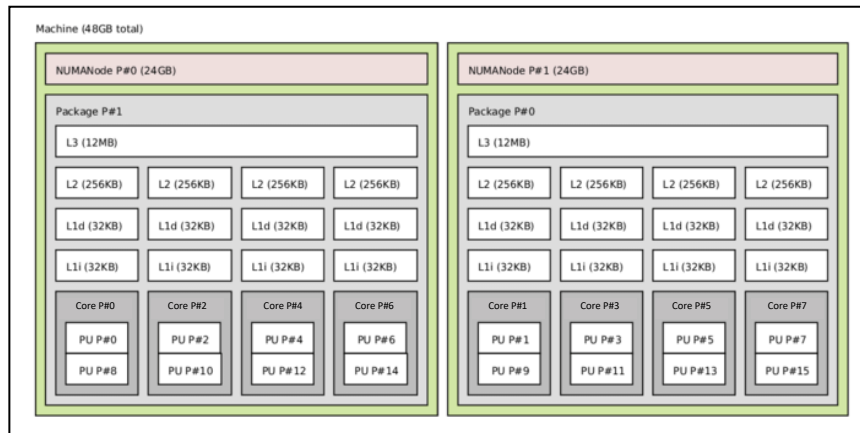
Describeix el teu entorn per fer l'examen:

- a) Ordinador amb Linux, i sistema corrent nadiu
- b) Ordinador amb Linux, executant-se en una màquina virtual
- c) Ordinador amb Windows, i subsistema Linux - Ubuntu
- d) Ordinador amb Windows, amb accés a un altre ordinador amb Linux (per ssh, putty o alguna altra eina de connexió per terminal)
- e) Ordinador amb Mac, sense accés a cap ordinador amb Linux
- f) Ordinador amb Mac, amb accés a un altre ordinador amb Linux (per ssh, o alguna altra eina de connexió per terminal)
- g) Altres: descriu el vostre entorn

Crea un fitxer “**respostes.txt**” per escriure les respostes a preguntes que no siguin de implementar codi.

### Exercici 1 (2,5 punts)

Ens passen la següent sortida de l'execució de la comanda lstopo en un:



A) Contesta **raonant breument** les teves respostes:

- 1) **(0,5 punts)** A dalt posa “48GB total”, però en la imatge també veiem (en la part rosa) “NUMANode” de “24 GB”. Per tant, podem afirmar que té un únic processador amb un total de 48GB de memòria?
- 2) **(0,5 punts)** A partir d'aquesta imatge, indica el nombre de hardware threads que té aquest sistema. Podem dir que aquest és el mateix nombre que el màxim de software threads que poden existir en el sistema?

B) Ara ens donen dos programes que són multithreaded i que executen 4 software threads cadascun. Els dos programes són intensos en ALU senzilla (sumes i restes). A partir de la informació de la imatge anterior i de lo que has vist en les sessions de laboratori, explica, **raonant breument** la teva resposta:

- 3) **(0,5 punts)** quina seria la distribució de software threads per aconseguir que el rendiment sigui el pitjor possible.
- 4) **(0,5 punts)** quina seria la distribució de software threads per aconseguir el millor rendiment.
- 5) **(0,5 punts)** per què pot ser possible que el temps real (o elapsed) que mostra les eines que hem vist en el laboratori per mesurar el temps d'execució d'un programa pot ser superior a la suma dels temps d'usuari i de sistema?

### Exercici 2 (5,5 punts)

**(2 punts)** Dins del codi associat a aquest control de laboratori, trobaràs els següents fitxers:

- coru.val - columna de valors amb el nivell del mar a La Coruña
- do-gen-plot.cpp - codi per transformar els valors a html
- Makefile - regles de compilació
- executa.sh - script per executar do-gen-plot sobre el fitxer de valors indicat
- executa.c - **fitxer que heu de completar per fer el mateix que executa.sh**

Completa el programa “**executa.c**” (està indicat en les línies “A”, “B”, i “C” en el codi) de forma que executi el programa do-gen-plot amb el codi de l'estació passat com a paràmetre. Per completar el codi segueix les indicacions d'aquests tres punts. Per validar-ho pots fer servir

`$ ./executa coru # per comprovar que funciona`

I amb el resultat, “coru.html”, podràs veure la gràfica de les alçades del mar usant un navegador com el Firefox.

Entre els fitxers adjunts, tenim el “coru.val” que està en format ASCII i que conté les dades del nivell marítim. Suposa que implementem un nou programa que llegeix aquest fitxer en format ASCII i genera per la sortida estàndard les dades en format binari. Indica:

- 1) **(0,5 punts)** quina línia de comandes has d'utilitzar per redirigir la sortida a un fitxer de dades, en lloc de mostrar-lo per pantalla.
- 2) **(1 punt)** quina relació de tamany existeix si comparem el tamany del fitxer en ASCII vs el del fitxer en format intern. Quin és el motiu? Si utilitzem la comanda “xxd” vista en alguna sessió del lab, com ho podríem validar?
- 3) **(0,5 punts)** si ens demanen fer aquests codis en Python, en lloc de C/C++, seria viable? En cas que sí, a priori, veuríem un rendiment similar, inferior o superior? Per què?
- 4) **(1 punt)** si decidíssim paral·lelitzar un codi que fes l'anàlisi de les dades per determinar els valors mínim, mitjana i màxim, amb OpenMP, quina línia de comandes s'hauria de introduir per fer aquesta compilació? A més, quina línia de comandes s'hauria de posar a l'hora d'executar el programa per configurar la variable d'entorn que determina el nombre de threads que volem utilitzar en el codi paral·lel (per exemple, 4)?
- 5) **(0,5 punts)** quina línia de comandes ens pot donar la informació de manera recurrent (cada poc temps i de manera automàtica) sobre els processos que estan en execució, el consum de memòria, els PIDs, estat en el que estan i prioritat de cada procés?

### Exercici 3 (2 punts)

- 1) **(0,5 punts)** Indica les dues línies de comandes per: (1) generar una versió compilada amb llibreries enllaçades estàticament; i (2) la línia de comandes per la versió compilada amb llibreries enllaçades dinàmicament, dels codis que ja us donem.
- 2) **(0,5 punts)** Indica quina línia de comandes hem d'utilitzar per poder obtenir la taula de símbols de qualsevol dels dos executables.
- 3) **(1 punt)** Per últim, indica quines diferències podem observar en la taula de símbols i quin és el motiu de mostrar el contingut que té cada versió.

### Entrega

Has de lliurar el fitxer “respostes.txt” i “executa.c”. Ho pots fer amb la següent línia de comandes:

```
$ tar czvf ProvaLab.tar.gz respostes.txt executa.c
```