



Search courses

Q (Search courses)



Commencé le	lundi 22 mars 2021, 15:40
État	Terminé
Terminé le	lundi 22 mars 2021, 16:00
Temps mis	19 min 34 s
Note	3,50 sur 14,00 (25%)

Question 1

Partiellement
correctNote de 0,67
sur 1,00

Pourquoi initialiser les éléments des tableaux et pourquoi le faire avec des valeurs pseudo-aléatoires déterministes (reproductibles d'un run à l'autre) dans des intervalles précis (par ex entre 0 et 2) ?

Veuillez choisir au moins une réponse :

- ☒ a. Pour qu'ils soient résidents en cache/mémoire. 
- ☐ b. Pour pouvoir comparer deux runs (notamment deux versions différentes d'une boucle).
- ☒ c. Pour éviter des exceptions flottantes (divisions par 0 etc.) ou des dépassements de capacités ("overflow"). 
- ☐ d. Pour permettre la vectorisation.

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 2.

Les réponses correctes sont : Pour qu'ils soient résidents en cache/mémoire., Pour éviter des exceptions flottantes (divisions par 0 etc.) ou des dépassements de capacités ("overflow")., Pour pouvoir comparer deux runs (notamment deux versions différentes d'une boucle).

Question 2

Correct

Note de 1,00
sur 1,00

A quoi servent les "méta-répétitions" (le fait de répéter plusieurs fois l'expérience complète) ? Si ça peut vous aider, réfléchissez à ce qui se passerait si on s'en passait ?

Veuillez choisir une réponse :

- ☐ a. Avoir des mesures plus stables (par ex écart type faible).
- ☐ b. Avoir des mesures plus précises (rapport entre temps mesuré et biais du timer).
- ☒ c. Evaluer/mesurer la stabilité (par ex via un écart type ou l'écart entre une médiane et un minimum) ✓

Votre réponse est correcte.

La réponse correcte est : Evaluer/mesurer la stabilité (par ex via un écart type ou l'écart entre une médiane et un minimum).



Question 3

Incorrect

Note de 0,00
sur 1,00

A quoi servent les répétitions de warmup (le fait de répéter le noyau avant le start) ? Si ça peut vous aider, réfléchissez à ce qui se passerait si on s'en passait ?

Veuillez choisir au moins une réponse :

- ☐ a. Charger les tableaux dans le niveau mémoire visé.
- ☒ b. Améliorer la précision des mesures (rapport entre mesure et biais du timer). 
- ☐ c. Passer le régime transitoire (pour ne mesurer que lors du régime permanent).
- ☒ d. Accélérer le processeur à sa fréquence nominale. 

Votre réponse est incorrecte.

Les réponses correctes sont : Charger les tableaux dans le niveau mémoire visé., Passer le régime transitoire (pour ne mesurer que lors du régime permanent)., Accélérer le processeur à sa fréquence nominale.

Question 4



Partiellement correct

Note de 0,50 sur 1,00

Comment accélérer (compilateur + code source) la boucle suivante (dont la plupart du temps est passé à diviser par $d[j]$) ? Une ou plusieurs bonnes réponses possibles.

```
// float a[], b[], c[], d[];  
for (i=0; i<n; i++)  
    a[i] = (b[i] / d[j]) + (c[i] / d[j]);
```

Veuillez choisir au moins une réponse :

- ☐ a. Remplacer la division par la multiplication par son inverse ($1/d[j]$).
- ☒ b. Factoriser la somme $\Rightarrow (b[i] + c[i]) / d[j]$ 
- ☒ c. Vectorisation. 
- ☐ d. Utiliser une option du compilateur (relative aux divisions).

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 2.

Les réponses correctes sont : Vectorisation., Remplacer la division par la multiplication par son inverse ($1/d[j]$)., Utiliser une option du compilateur (relative aux divisions)., Factoriser la somme $\Rightarrow (b[i] + c[i]) / d[j]$

Question 5

Incorrect

Note de 0,00
sur 1,00

A quoi reconnaissez-vous une boucle vectorisée ?

Veuillez choisir une réponse :

- ☒ a. En x86 et pour des flottants simple/double précision, son code assembleur utilise des instructions suffixées par SS/SD. ✗
- ☐ b. En x86 et pour des flottants simple/double précision, son code assembleur utilise des instructions suffixées par PS/PD.
- ☐ c. Il est impossible de le savoir même au niveau assembleur.
- ☐ d. Son code source manipule des tableaux.
- ☐ e. Son code source est régulier et simple.

Votre réponse est incorrecte.

La réponse correcte est : En x86 et pour des flottants simple/double précision, son code assembleur utilise des instructions suffixées par PS/PD.

Question 6

Correct

Note de 1,00
sur 1,00

Quelle est la meilleure façon d'optimiser la boucle suivante (on considère que les options permettant la vectorisation sont présentes et que f, g et h sont vectorisables) ?

```
// n%4 == 0
for (i=0; i<3*n; i++) {
    if (i < n) f(i);
    else if (i < 2*n) g(i);
    else h(i);
}
```

Veuillez choisir une réponse :

- ☐ a.

```
for (i=0; i<3*n; i+=4) {
    if (i < n) {
        f(i); f(i+1); f(i+2); f(i+3);
    } else if (i < 2*n) {
        g(i); g(i+1); g(i+2); g(i+3);
    } else {
        h(i); h(i+1); h(i+2); h(i+3);
    }
}
```
- ☐ b.

```
for (i= 0; i< n; i+=4) { f(i); f(i+1); f(i+2); f(i+3); }
for (i= n; i<2*n; i+=4) { g(i); g(i+1); g(i+2); g(i+3); }
for (i=2*n; i<3*n; i+=4) { h(i); h(i+1); h(i+2); h(i+3); }
```
- ☒ c.

```
for (i=0; i<n; i++) f(i);
for (i=n; i<2*n; i++) g(i);
for (i=2*n; i<3*n; i++) h(i);
```

 ✓

Votre réponse est correcte.

La réponse correcte est :

```
for (i=0; i<n; i++) f(i);
for (i=n; i<2*n; i++) g(i);
for (i=2*n; i<3*n; i++) h(i);
```

Question 7

Incorrect

Note de 0,00
sur 1,00

Comment la localité temporelle est elle exploitée par le matériel ?

Une ou plusieurs bonnes réponses.

Veuillez choisir au moins une réponse :

- ☐ a. Caches (de données, d'instructions et TLB)
- ☐ b. Cachelines (lignes de caches, fait de charger les données par bloc plutôt qu'octet par octet).
- ☐ c. Hardware prefetchers.
- ☐ d. Pipeline.
- ☒ e. Elle n'est pas exploitée par le matériel.



Votre réponse est incorrecte.

La réponse correcte est : Caches (de données, d'instructions et TLB)

Question 8

Incorrect

Note de 0,00
sur 1,00

Qu'est-ce que la localité spatiale ?

Veuillez choisir une réponse :

- ☐ a. Réutilisation des accès (adresses identiques)
- ☐ b. Une optimisation du compilateur
- ☐ c. Propriété qu'ont les prochains accès à être regroupés (adresses proches)
- ☒ d. Une optimisation du processeur



Votre réponse est incorrecte.

La réponse correcte est : Propriété qu'ont les prochains accès à être regroupés (adresses proches)

Question 9

Partiellement
correctNote de 0,17
sur 1,00

Parmi les runs suivants, lesquels seront assez longs pour que les profils (coût des différentes fonctions/boucles) soit suffisamment fiable/précis (au moins 500 échantillons).

Par défaut MAQAO échantillonne à 200 Hz mais on peut le pousser à 1000 Hz avec lprof_params="sampling-rate=high".

Veuillez choisir au moins une réponse :

- ☒ a. 0.05s à 1000 Hz
- ☒ b. 0.52s à 1000 Hz
- ☐ c. 0.52s à 200 Hz
- ☐ d. 2.6s à 200 Hz
- ☐ e. 2.4s à 200 Hz



Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 1.

Les réponses correctes sont : 0.52s à 1000 Hz, 2.6s à 200 Hz

Question 10


Incorrect

Note de 0,00
sur 1,00

Quelle est la meilleure clause pour équilibrer la boucle suivante ?

```
#pragma omp parallel for  
for (i=0; i<n; i++)  
    a[i] = b[i] + c[i];
```

Veuillez choisir une réponse :

- ☒ a. 
- ☐ b.
- ☐ c.

Votre réponse est incorrecte.

La réponse correcte est :

Question 11

Partiellement
correct

Quelles actions peuvent impacter la durée nécessaire (nb de répétitions) de warmup (donc le profil du régime transitoire) ?

[Obtenir l'app mobile](#)