

M3 L3: Bande-passante, stockage et réseau

Remarque : en l'absence de précision, la taille d'un mot est 4 octets

1. Bande-passante d'un ordinateur

Vous êtes dans un magasin pour acheter un ordinateur. Vous hésitez entre deux possibilités :

- L'ordinateur A a un processeur à 2 GHz et un disque dur ayant une bande passante de 400 Mo/s. Il a 8 Mo de cache ;
- l'ordinateur B a un processeur à 3 GHz et et un disque dur ayant une bande passante de 12.5 MegaBlocs/s (chaque bloc se compose de 4 mots). Il a 4 Mo de cache.

Que choisissez vous si vous désirez minimiser le temps nécessaire pour :

- lire un fichier ?
- exécuter un programme qui accède souvent à la mémoire ? (Remarquer que les caches de A et B sont du même modèle, mais il y a deux fois plus de blocs dans celui de A que dans celui de B.)

2. Bande-passante bis

Vous hésitez entre deux choix d'ordinateurs. Vous demandez plus de détails et découvrez que pour l'ordinateur A, chaque accès au cache prend 1 ns (=1 nanoseconde) et chaque défaut de cache prend 120 ns, tandis que pour l'ordinateur B, chaque accès au cache prend 1.2 ns et chaque défaut de cache prend 100 ns. Vous décidez de prendre un exemple d'un programme réel pour comparer les deux ordinateurs. L'algorithme du programme envisagé est le suivant :

Algorithme pour programme de test
entrée : x entier naturel sortie : y entier naturel
<pre> $y \leftarrow 1$ Si $x = 0$ <i>ou</i> $x = 1$ $y \leftarrow 0$ $s \leftarrow 0, i \leftarrow 2$ Tant que $s \leq x$ <i>et</i> $y = 1$ Si i divise x $y \leftarrow 0$ Sinon $i \leftarrow i + 1$ $s \leftarrow i \times i$ Sortir : y </pre>

Supposons que pour chaque ordinateur la mémoire cache contient 2 blocs et la mémoire en contient 4. Chaque bloc contient 4 mots. L'adresse d'un mot en mémoire est dans l'intervalle [0, 15].

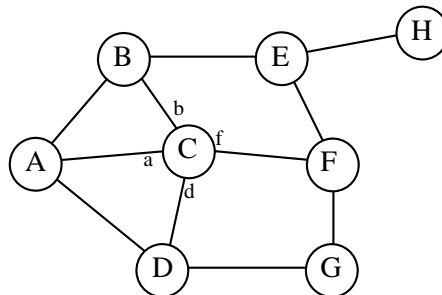
Dans cet exercice on ne compte pas les accès mémoire nécessaires pour lire les instructions. On suppose par ailleurs que le processeur a plus de 4 registres de sorte qu'avec le programme utilisé ici, une fois qu'une variable a été lue dans le cache il n'est pas nécessaire de la relire.

Supposons enfin que x est stocké à l'adresse mémoire 0, y en 1, i en 12, et s à l'adresse 13 (toutes les variables sont initialement en mémoire), et que la valeur de x est initialement 4.

- Que fait le programme ci-dessus ?
- Explicitez les accès à la mémoire que le programme fait pour lire les 4 variables.
- Combien d'accès au cache a-t-on ? Combien de défauts de cache a-t-on ?
- Si vous devez choisir l'ordinateur le plus rapide, lequel prendriez-vous sur la base de cet algorithme ?

3. Routage IP

Considérez le graphe de routeurs Internet suivant :

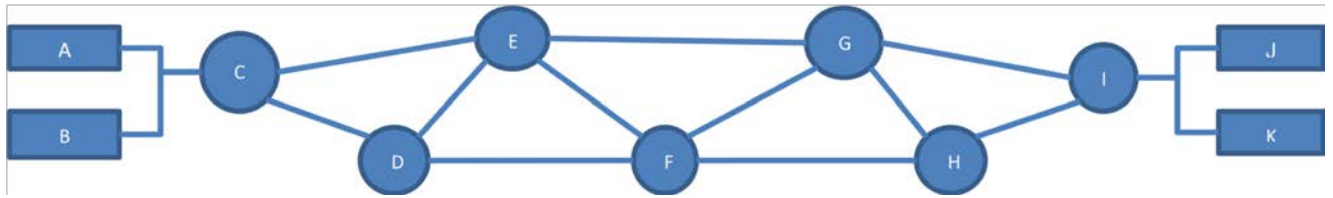


- Simulez l'algorithme de mise à jour de table de routage par « oui-dire » en partant du nœud A. Notez sur chaque connexion les longueurs de chemin vers le nœud A communiquées par chaque nœud à ses voisins (ne retenez que le plus court). Biffez les longueurs non retenues.
- Donnez la table de plus court chemin du nœud C vers tous les autres en indiquant à chaque fois la longueur totale et la première connexion à suivre (cf exemple vers A donné dans la table) :

table de C

A	1	a
B		
D		
E		
F		
G		
H		

4 Routage encore: Considérez le graphe de routeurs Internet suivant :



a) Remplissez les informations manquantes dans les extraits des tables de routage suivantes indiquant les chemins vers A et K des routeurs C à I.

C			D			E		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A		
K			K			K		

F			G			H			I		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A			A		
K			K			K			K		

b) Comment ces extraits doivent-ils être modifiés si le lien entre E et G est rompu?

C			D			E		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A		
K			K			K		

F			G			H			I		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A			A		
K			K			K			K		

c) Comment ces extraits doivent-ils être modifiés si le routeur E tombe en panne?

C			D			E		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A		
K			K			K		

F			G			H			I		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
A			A			A			A		
K			K			K			K		

5 Routage toujours

On considère un réseau dans lequel se trouvent plusieurs routeurs (A, B, C, ..., N). On connaît en partie les tables de routage des routeurs A, F et H qui sont :

A			F			H		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
B	C	2	J	O	2	D	I	2
D	C	5	L	K	2	C	B	2
N	x	y	M	N	2	F	J	3

En se basant uniquement sur un strict minimum de liens entre routeurs qui doivent exister selon les tables partielles ci-dessus (i.e. sans en imaginer d'autres non-justifiés par ces tables), quelles sont les valeurs de x et y ?

6 Encore un peu de routage?

On considère une topologie réseau comprenant 6 routeurs nommés A, B, C, D, E, F. On connaît leurs tables de routages :

A			B			C		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
C	B	2	D	C	2	F	E	2
E	F	2	F	A	2	A	B	2
D	B	3	E	C	2			
D			E			F		
dest.	dir.	dist.	dest.	dir.	dist.	dest.	dir.	dist.
B	C	2	D	C	2	B	A	2
E	C	2	A	F	2	C	E	2
F	C	3	B	C	2	D	E	3
A	C	3						

Le routeur B tombe en panne et n'est plus utilisable. Tous les autres routeurs sont avertis de cette panne et leur tables de routages sont mises à jour de façon à éviter le routeur B.

Existe-t-il encore une route de A à D après la panne; et si oui quelle est sa longueur?

- le répertoire home est un fichier contenant les tables de ses fichiers et répertoires, dont celle du répertoire muche — ...
- le répertoire Musique est un fichier contenant les tables de ses fichiers et répertoires, dont celle de notre fichier (donnée plus haut).

Si l'on suppose que notre fichier est le seul dans le répertoire **Musique**, on va y trouver seulement la *table donnant l'adresse des blocs de ce fichier*. Combien de blocs utilise alors ce répertoire Musique sachant que la table commence par indiquer le nombre de blocs occupés par le fichier et qu'une adresse occupe 32 bits pour cet exercice ?

En supposant que les blocs du répertoire Musique sont stockés à partir de l'adresse **42**, et en supposant que le répertoire **muche** ne contient que le répertoire Musique, quel est le début du fichier représentant le répertoire muche (donner le contenu binaire de sa table des adresses des blocs du répertoire muche) ?

7.6 Décodage du « message » de départ

Sachant

- que les adresses indiquées ci-dessous sont au format : « **adresses de bloc.adresse de mot** », avec un bloc de taille **4Kio** et des **mots de 32 bits**

- que le disque contient les informations suivantes :

```

...
42.0: 0000000000000000000000000000000011110 // table des adresses, nb de blocs = 30
42.1: 00000000000000000000000000000000101011
...
42.1023: 11101011101010010010001001001110 // amplitude codée avec code de Huffman
43.0: 11001100110101010100001110001001
43.1: 00110100100110001001000111001011
...

```

- que le code de Huffman utilisé est le suivant (les quelques valeurs ci-dessous nous suffisent) :

valeur sur 16 bits (décimale)	valeur sur 16 bits (binaire)	Code de Huffman
...		
-26922	1110100100101010	01010101010101
...		
0	0000000000000000	11001100110
...		
5929	0001011100101001	10101010
...		
11056	0010101100110000	0001110001
...		
26922	0110100100101010	11111111111111
...		

Quelle est la valeur décimale de l'amplitude représentée sur 16 bits pour notre signal au temps $2/44100$ s sachant que la mesure commence au temps 0s ?

→ pour cela, utiliser l'adresse trouvée à la question 7.4 et décoder les codes de huffman jusqu'à celui correspondant au 3^{ème} échantillon.

L'amplitude maximum sur 16 bits correspond en fait à la valeur 4 ; sachant cela, on peut vérifier qu'on obtient bien la valeur de la somme donnée en 3.2 pour le temps $2/44100$.