Dea d'Informatique
Année 97-98
Algorithmes et Combinatoire
Robert Cori

Réseaux de tri

On peut considérer un tri comme une séquence de transpositions. Les *réseaux de tri* n'utilisent que des comparateurs qui prennent en entrée deux nombres a et a' et donnent en sortie $\min(a, a')$ et $\max(a, a')$. En C, un comparateur se représente par la procédure

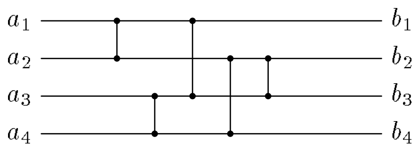
```
void C(int i, int j)
{
    int     x;

    if( a[i] > a[j] ) {
        x = a[i]; a[i] = a[j]; a[j] = x;
    }
}
```

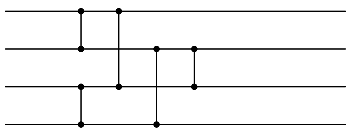
Graphiquement, le fragment de programme

```
C(1,2); C(3,4); C(1,3); C(2,4); C(2,3)
```

peut se représenter par le réseau ci-dessous. Les entrées sont notées a_1, a_2, a_3, a_4, les sorties b_1, b_2, b_3, b_4. On suppose donc que les données circulent de la gauche vers la droite dans un tel réseau, et que chaque comparateur est représenté par une connexion verticale entre 2 lignes.



Les réseaux autorisent que 2 comparaisons puissent se faire simultanément. Par exemple, le réseau précédent équivaut au réseau.



Quoique difficile à dessiner, les étapes 2 et 3 peuvent aussi se faire en parallèle. Ce réseau a donc un *délai* de 3.

1- Montrer que le réseau précédent trie bien ses entrées.

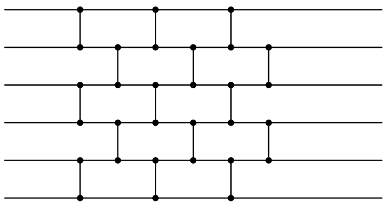
2- Dessiner les réseaux pour les tris par bulle et par insertion. Que constate-t-on?

3- Un réseau est *primitif* s'il ne fait que des comparaisons entre fils adjacents, c'est à dire des opérations `C(i,i+1)`. Montrer que tout réseau primitif a besoin d'au moins $n(n-1)/2$ comparateurs. (On pourra raisonner sur le nombre d'inversions, c'est à dire de couples (i,j) tels que $i < j$ et $a_i > a_j$).

4- Montrer que pour dans un réseau primitif, si on a en sortie $b_i > b_j$ pour $i < j$ et pour un vecteur a de n entrées (n lignes), on a aussi $b_i > b_j$ si le vecteur d'entrée est la suite $\langle n, n-1, \cdots, 2, 1\rangle$. (On pourra raisonner par récurrence sur le nombre de comparateurs du réseau).

5- Cette propriété est-elle aussi vraie dans un réseau non primitif?

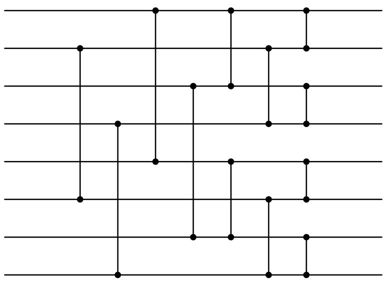
6- Montrer que le réseau suivant *pair-impair* (à n étages) fait bien le tri de ses entrées. Quel est son délai?



Ce tri correspond au programme C

```
for (k = 1; k <= n; ++k)
    if (k % 2 != 0)
        for (j = 1; j < n; j = j + 2)
            C(j, j+1);
    else
        for (j = 0; j < n; j = j + 2)
            C(j, j+1);
```

7- Une suite $\langle a_1, a_2, \cdots, a_n\rangle$ est *bitone* s'il existe un k tel que $a_1 \geq a_2 \geq \cdots \geq a_k \leq a_{k+1} \leq a_{k+2} \leq \cdots \leq a_n$. On remarque que toute sous-séquence d'une suite bitone est bitone. Montrer que pour $n = 2^k$, le réseau suivant met en ordre croissant toute suite bitone. Avec quel délai?



7- Si on sait inverser un réseau de tri en polarisant les comparateurs, montrer que le tri par fusion peut se faire avec le réseau précédent. Quel est le délai d'un tel tri?