

Preuve et Analyse des Algorithmes LSPI-3 informatique Toulon.

27 juin 2014

Le sujet est composé quatre exercices indépendants à traiter en moins de deux heures. Aucun document autorisé. Vous êtes invités à remettre une copie claire, concise, sans rature ni surcharge. Il est par ailleurs inutile de recopier l'énoncé. . . La note finale tiendra compte de la présentation générale de la copie.

Q 1. On note F_n le n -ième terme de la suite de Fibonacci de termes initiaux $F_0 = 0$, $F_1 = 1$. On rappelle que

$$F_n = \frac{1}{\sqrt{5}}(\phi^n - \hat{\phi}^n)$$

où $\phi > \hat{\phi}$ désignent les deux racines du polynôme $T^2 - T - 1$.

1. Calculer ϕ et $\hat{\phi}$.
2. Calculer $\lim_{n \rightarrow \infty} \phi^n$, et $\lim_{n \rightarrow \infty} \hat{\phi}^n$.

Lesquelles de ces affirmations sont correctes

1. $F_n = O(5^n)$,
2. $F_n = \Omega(n!)$,
3. $F_n \sim \frac{1}{\sqrt{5}}\phi^n$,
4. $F_n = o(n^5)$,

Q 2.

1. Déterminer une solution de $24149u + 28681v = 1$.
2. Quel est l'inverse de 24149 modulo 28681 ?

Q 3. On considère la fonction récursive **f** de la figure fig. 1 .

1. Que calcule cette fonction ?
2. Estimer le temps de calcul en fonction de n .
3. Préciser le domaine de validité.

```
long long f( int )
{
if ( n > 1 ) return f(n-1) + f(n-2);
return n;
}
```

FIG. 1 – code C de la fonction f .

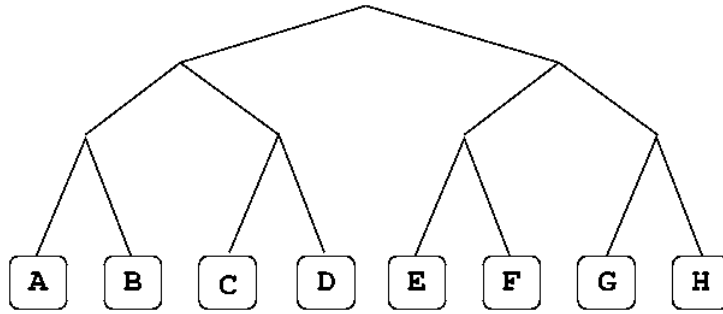


FIG. 2 – Comment déterminer les 2 plus petites valeurs en 9 comparaisons

4. Existe-t-il une meilleure manière d'implanter cette fonction ?

Q 4. Etant donné un tableau T de n nombres rationnels, un entier $1 \leq k < n$, on s'intéresse à la détermination des k valeurs minimales de T . Pour simplifier, nous supposons d'une part que n est une puissance de 2, et d'autre part que les valeurs de T sont deux à deux distinctes. Il s'agit d'écrire un algorithme pour permutationner les éléments de T de sorte à obtenir :

$$\forall i \quad 0 < i < k, \quad \forall j, \quad k \leq j < n, \quad T[i] \leq T[j].$$

1. Combien faut-il faire de comparaisons pour déterminer la plus petite valeur d'un tableau de taille n ?
2. Ecrire une fonction naïve `void min(int *T, int k, int n)` qui place les k plus petites valeurs de T sur dans les k premières positions.
3. Exprimer le nombre de comparaisons en fonction de k et de n .
4. Quand $\log n$ devient négligeable devant k , la solution naïve n'est pas optimale. Pourquoi ?
5. Comment déterminer les 2 plus petites valeurs du tableau de taille 8 (fig. 2) en moins de 9 comparaisons.
6. Expliquer, sans chercher à écrire d'algorithme, comment obtenir les 2 plus petites valeurs de T en $n - \log(n) - 2$ comparaisons.

