جامعة هواري بومدين للعلوم و التكنولوجيا
Université des Sciences et de la Technologie Houari Boumediene
Faculté d'Electronique et d'Informatique
Département d'Informatique

Concours d'accès au Doctorat 3 ième Cycle Informatique 2019– 2020

Le 26/10/2019

Matière 1 : Algorithmique avancée et complexité, Systèmes d'exploitation,
Coefficient **1**, durée **1h30**
(Spécialité : RSI)

Partie 1 : Algorithmique avancée et complexité

Exercice 1 : (8 points)

a. Donnez les définitions des notations Landau O, Θ et Ω.
a. Les affirmations ci-dessous sont-elles vraies ou fausses ? Si vous pensez qu'une affirmation est fausse, indiquez pourquoi et corrigez l'affirmation. (Reproduire, sur la copie, le tableau ci-dessous et le compléter)

Affirmation	Vraie ou Fausse?	Affirmation correcte
Si $f(n) \in O(g(n)),\ alors\ g(n) \in O(f(n))$.		
Si $f(n) \in O(g(n)), alors\ f(n) + g(n) \in O(g(n))$		
La meilleure borne asymptotique de $O(f(n)) + O(g(n))$ est $O(f(n) + g(n))$		
Si $g = O(f(n))$ et $h = O(g(n))$ alors $h = O(f(n))$		
$5n + 8n^2 + 100n^3 \neq O(n^4)$		
$100n + \log n = \Theta(n + (logn)^2)$		
$n2^n = \Theta(3^n)$		

Exercice 2 : (12 points)

Une attaque par déni de service (DOS) est une attaque visant à saturer une application de sorte qu'elle ne puisse plus répondre à de nouvelles requêtes. Typiquement, un pirate essaiera de saturer l'application par une multiplication de requêtes lourdes en temps de calcul. Un exemple classique est celui de "Apache" qui, en 1997, intégrait la fonction suivante :

```
Procédure Proc1(E-S/ T : Tableau[n] de caractère ; E/ n : entier) ;
Var x, y : entier ;
Début
   x←2 ;
   tant que (x<=n) Faire
         Si (T[x-1] = '/') et (T[x] = '/') Alors
               Pour y ←x+1 à n Faire    T[y-1] ← T[y] ;      fait;
               n←n-1 ;
         Sinon x ←x+1 ;
         finsi ;
   fait;
Fin;
```

a. Que fait cette fonction ?
b. Quelle est sa complexité dans le pire des cas ?
c. Comment pouvait-on donc faire pour saturer un serveur web Apache ?
d. Le patch proposé pour corriger cette faille de sécurité implémentait no2slash de la façon ci-après. Quelle est la complexité de cette seconde fonction ? La faille est-elle corrigée ?

```
Procédure Proc1(E-S/ T : Tableau[n] de caractère ; n : entier) ;
Var x, y : entier ;
Début
    x ←1 ;    y ←1;
    Tant que (x <=n) Faire
        T[y]←T[x] ;
        Si (T[y] = '/') Alors
                Tant que (x<=n) et (T[x]= '/') Faire
                        x ← x+1;
                Fait ;
        Sinon x ←x+1 ;
        Finsi ;
        y←y+1 ;
    Fait;
    T[y] ←  '\0'  /* caractère de fin de chaine*/
Fin;
```

Bon courage