

IF 2

QCM 3, Version: A

Nom: _____

Carte d’étudiant: _____

Remplissez la table avec les lettres correspondant à vos réponses.

Questions	1	2	3	4	5	6	7	8	9	10	11	12	13
Réponse(s)													

Bonne réponse=1pt; mauvaise réponse ou réponse incomplète =-0,25pt; pas de réponse=0pt

1. Pour le morceau de programme: `if (a>b) m=a; else m=b;` et la pré-condition : $a = A$ et $b = B$ (où A et B sont des constantes positives), parmi les assertions suivantes laquelle n’est **pas** une post-conditions:

- (a) $m = \max(A, B)$
- (b) $1 \neq 0$
- (c) $m \geq |A - B|$
- (d) $A > B$

2. On considère la boucle: `for(int i=0; i<n; i+=3) f(i);` Trouver la bonne réponse:

- (a) le nombre d’appels de f est un $\Theta(n)$
- (b) le nombre d’appels de f est un $\Theta(\log(n))$
- (c) le nombre d’appels de f est un $\Theta(10^n)$

3. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si cet arbre a n feuilles, le nombre total de noeuds (attention: les feuilles sont des noeuds) est

- (a) $2n - 1$
- (b) 2^n
- (c) $2 \log(n)$

4. On considère le programme suivant:

```
static boolean estPresent(int v,int[]t, int l,int r){
    if(l>r) return false; int m=(l+r)/2;
    if(v==t[m])return true;
    if(v<t[m])return estPresent(v,t,l,m-1); else return estPresent(v,t,m+1,r);}
```

Ce programme retourne true si et seulement si v est présent dans t entre les indices l et r en (choisir la (ou les) bonne(s) réponse(s):

- (a) $\Theta(n)$ comparaisons dans le pire cas
- (b) $\Theta(\log(n))$ comparaisons dans le pire cas
- (c) $\Theta(n)$ comparaisons dans le meilleur cas

5. Le tri par insertion tri un tableau de n éléments en:

- (a) $\Theta(n^2)$ comparaisons dans le pire cas
- (b) $\Theta(n \log(n))$ comparaisons dans le pire cas
- (c) $\Theta(n)$ comparaisons dans le pire cas

6. On considère la boucle suivante: `for(int i=1; i<n; i=i*10) f(i);` Parmi les assertions suivantes laquelle est FAUSSE:

- (a) le nombre d’appels de f est un Θ du nombre n
- (b) le nombre d’appels de f est un Θ du nombre de bits nécessaires pour représenter n en binaire
- (c) le nombre d’appels de f est un Θ du nombre de chiffres dans la représentation de n en base 10

7. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si toutes les branches ont la longueur n , le nombre de feuilles est:
- 2^n
 - n
 - $\log(n)$
8. On considère la boucle suivante: `for(int i=0; i<n; i+=1) for(int j=1; j<n; j=2*j) f(i);` Trouver la bonne réponse:
- le nombre d'appels de f est un $\Theta(n)$
 - le nombre d'appels de f est un $\Theta(n^2)$
 - le nombre d'appels de f est un $\Theta(\log(n))$
 - le nombre d'appels de f est un $\Theta(n \log(n))$
9. On considère la boucle: `for(int i=n; i>1; i/=10) f(i);` Trouver la bonne réponse:
- le nombre d'appels de f est un $\Theta(n)$
 - le nombre d'appels de f est un $\Theta(n^2)$
 - le nombre d'appels de f est un $\Theta(\log(n))$
 - le nombre d'appels de f est un $\Theta(n \log(n))$
10. On considère la boucle: `for(int i=0; i<n; i+=1) f(i);` Trouver la bonne réponse:
En fonction du *nombre k de chiffres* dans la représentation de n :
- le nombre d'appels de f est un $\Theta(k)$
 - le nombre d'appels de f est un $\Theta(\log(k))$
 - le nombre d'appels de f est un $\Theta(10^k)$
11. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si cet arbre a n feuilles, toutes les branches ont la longueur:
- 2^n
 - n
 - $\log(n)$
12. On considère le morceau de programme: `tmp=t[0]; for(int i=1; i<t.length; i++)if(tmp<t[i])tmp=t[i];`
Choisir parmi les assertions suivantes celle qui est vraie:
- " tmp est égal au max du tableau t entre les indices 0 et $i - 1$ " est un invariant de la boucle
 - " tmp est égal au max du tableau t entre les indices 0 et $t.length$ " est un invariant de la boucle
 - il n'y a pas d'invariant pour cette boucle
13. On considère le programme suivant:
- ```
static boolean cherche(int v, int[] t, int l, int r)
{for(int i=l; i<=r; i++)if(t[i]==v)return true; return false; }
```
- Ce programme retourne `true` si et seulement si  $v$  est présent dans  $t$  entre les indices  $l$  et  $r$  en : (choisir la (ou les) bonne(s) réponse(s))
- $\Theta(n)$  comparaisons dans le pire cas
  - $\Theta(\log(n))$  comparaisons dans le pire cas
  - $\Theta(n)$  comparaisons dans le meilleur cas

# Answer Key for Exam A

Bonne réponse=1pt; mauvaise réponse ou réponse incomplète =-0,25pt; pas de réponse=0pt

1. Pour le morceau de programme: `if (a>b) m=a; else m=b;` et la pré-condition :  $a = A$  et  $b = B$  (où  $A$  et  $B$  sont des constantes positives), parmi les assertions suivantes laquelle n'est **pas** une post-conditions:

- (a)  $m = \max(A, B)$
- (b)  $1 \neq 0$
- (c)  $m \geq |A - B|$
- (d)  $A > B$

2. On considère la boucle: `for(int i=0; i<n; i+=3) f(i);` Trouver la bonne réponse:

- (a) le nombre d'appels de  $f$  est un  $\Theta(n)$
- (b) le nombre d'appels de  $f$  est un  $\Theta(\log(n))$
- (c) le nombre d'appels de  $f$  est un  $\Theta(10^n)$

3. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si cet arbre a  $n$  feuilles, le nombre total de noeuds (attention: les feuilles sont des noeuds) est

- (a)  $2n - 1$
- (b)  $2^n$
- (c)  $2 \log(n)$

4. On considère le programme suivant:

```
static boolean estPresent(int v,int[]t, int l,int r){
 if(l>r) return false; int m=(l+r)/2;
 if(v==t[m])return true;
 if(v<t[m])return estPresent(v,t,l,m-1); else return estPresent(v,t,m+1,r);}

```

Ce programme retourne true si et seulement si  $v$  est présent dans  $t$  entre les indices  $l$  et  $r$  en (choisir la (ou les) bonne(s) réponse(s):

- (a)  $\Theta(n)$  comparaisons dans le pire cas
- (b)  $\Theta(\log(n))$  comparaisons dans le pire cas
- (c)  $\Theta(n)$  comparaisons dans le meilleur cas

5. Le tri par insertion tri un tableau de  $n$  éléments en:

- (a)  $\Theta(n^2)$  comparaisons dans le pire cas
- (b)  $\Theta(n \log(n))$  comparaisons dans le pire cas
- (c)  $\Theta(n)$  comparaisons dans le pire cas

6. On considère la boucle suivante: `for(int i=1; i<n; i=i*10) f(i);` Parmi les assertions suivantes laquelle est FAUSSE:

- (a) le nombre d'appels de  $f$  est un  $\Theta$  du nombre  $n$
- (b) le nombre d'appels de  $f$  est un  $\Theta$  du nombre de bits nécessaires pour représenter  $n$  en binaire
- (c) le nombre d'appels de  $f$  est un  $\Theta$  du nombre de chiffres dans la représentation de  $n$  en base 10

7. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si toutes les branches ont la longueur  $n$ , le nombre de feuilles est:

- (a)  $2^n$
- (b)  $n$
- (c)  $\log(n)$

8. On considère la boucle suivante: `for(int i=0; i<n; i+=1) for(int j=1; j<n; j=2*j) f(i);` Trouver la bonne réponse:
- (a) le nombre d'appels de `f` est un  $\Theta(n)$
  - (b) le nombre d'appels de `f` est un  $\Theta(n^2)$
  - (c) le nombre d'appels de `f` est un  $\Theta(\log(n))$
  - (d) le nombre d'appels de `f` est un  $\Theta(n \log(n))$
9. On considère la boucle: `for(int i=n; i>1; i/=10) f(i);` Trouver la bonne réponse:
- (a) le nombre d'appels de `f` est un  $\Theta(n)$
  - (b) le nombre d'appels de `f` est un  $\Theta(n^2)$
  - (c) le nombre d'appels de `f` est un  $\Theta(\log(n))$
  - (d) le nombre d'appels de `f` est un  $\Theta(n \log(n))$
10. On considère la boucle: `for(int i=0; i<n; i+=1) f(i);` Trouver la bonne réponse:  
En fonction du *nombre  $k$  de chiffres* dans la représentation de  $n$ :
- (a) le nombre d'appels de `f` est un  $\Theta(k)$
  - (b) le nombre d'appels de `f` est un  $\Theta(\log(k))$
  - (c) le nombre d'appels de `f` est un  $\Theta(10^k)$
11. Un arbre binaire parfait est un arbre dont tous les noeuds qui ne sont pas des feuilles ont exactement deux fils et toutes les branches (chemin de la racine à une feuille) ont la même longueur. Si cet arbre a  $n$  feuilles, toutes les branches ont la longueur:
- (a)  $2^n$
  - (b)  $n$
  - (c)  $\log(n)$
12. On considère le morceau de programme: `tmp=t[0]; for(int i=1; i<t.length; i++)if(tmp<t[i])tmp=t[i];`  
Choisir parmi les assertions suivantes celle qui est vraie:
- (a) "`tmp` est égal au max du tableau  $t$  entre les indices 0 et  $i - 1$ " est un invariant de la boucle
  - (b) "`tmp` est égal au max du tableau  $t$  entre les indices 0 et  $t.length$ " est un invariant de la boucle
  - (c) il n'y a pas d'invariant pour cette boucle
13. On considère le programme suivant:
- ```
static boolean cherche(int v, int[] t, int l, int r)
{for(int i=l; i<=r; i++)if(t[i]==v)return true; return false; }
```
- Ce programme retourne `true` si et seulement si v est présent dans t entre les indices l et r en : (choisir la (ou les) bonne(s) réponse(s))
- (a) $\Theta(n)$ comparaisons dans le pire cas
 - (b) $\Theta(\log(n))$ comparaisons dans le pire cas
 - (c) $\Theta(n)$ comparaisons dans le meilleur cas