

IF 2

QCM final, Version: A

Nom: _____

Carte d’étudiant: _____

Remplissez les tables avec les lettres correspondant à vos réponses.

Réponses Section 1 (Questions générales):

section 1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Réponse(s)															

Réponses Section 2 (Invariants et complexité):

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

Réponses Section 3 (Autres questions):

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

Section 1. Questions générales

1. Pour la classe définie comme suit:

```
public class Bidon {int x;};
```

lequel des programmes est faux (ne peut s'exécuter)?

- (a) Bidon a=new Bidon(3);
- (b) Bidon a=new Bidon(); String s=a.toString();
- (c) Bidon a=new Bidon(); boolean b= a.equals("bonjour");
- (d) Bidon a=new Bidon(); boolean b= a.equals(a);

2. On considère la classe définie par :

```
class A {int i; static void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=a; a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

3. On considère la classe définie par : class B {int i=0; }

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a,b; a.i=10; b=a; b.i=5; System.out.println(a.i);
```

- (a) il affiche 10
- (b) il affiche 5
- (c) il provoque une erreur

4. On considère la classe définie par : class B {int i; }

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B();B b=new B(); a.i=10; b=a; b.i=5;  
if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");
```

- (a) il affiche EGAL
- (b) il affiche PAS EGAL
- (c) il provoque une erreur

5. On définit la méthode permuter par:

```
public static void permuter (String s1, String s2, int x1, int x2){  
    String tmp1=s1; s1=s2; s2=tmp1;  
    int tmp2=x1; x1=x2; x2=tmp2;  
}
```

On l'applique dans le contexte suivant:

```
String a="un"; String b="deux"; int c=3; int d =4; permuter(a,b,c,d);
```

Quelles seront les valeurs de a,b,c,d après l'exécution de ce code?

- (a) "un", "deux", 3, 4
- (b) "deux", "un", 3, 4
- (c) "un", "deux", 4, 3
- (d) "deux", "un", 4, 3

6. On considère la classe définie par :

```
class A {static int i; void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

7. On considère la classe définie par : `class B {int i; }` Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B(); B b=new B(); b.i=10; b=a; b.i=5; System.out.println(a.i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

8. Pour la classe définie comme suit:

```
public class Bidon {int x=5;};
```

Parmi les programmes suivants le(s)quel(s) provoque(nt) une erreur?

- (a) `Bidon a=new Bidon(3);`
- (b) `Bidon a=new Bidon(); String s="bidon"+a;`

9. On considère la classe définie par :

```
A {static int i; static void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

10. On considère la classe définie par : `class C {static int i; }`

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
C a=new C(); C b=new C(); a.i=10; b.i=5; System.out.println(a.i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

11. Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
int i,j; i=10; j=i; j=5; System.out.println(i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

12. Pour la classe D définie comme suit:

```
class D {  
    int x;  
    D() {x=3; };  
    D( int a){this(); x=x+a;};  
    D( int a, int b){this(b); x= x-a;}}
```

qu'affichera le code suivant?

```
D a=new D(5,6);  
System.out.println(a.x);
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4

13. On considère la classe définie par : `class B {int i; }`

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B(); B b=new B(); a.i=10; b.i=10;  
if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");
```

- (a) il affiche EGAL
- (b) il affiche PAS EGAL
- (c) il provoque une erreur

14. On considère la classe définie par :

```
class A {int i; void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

15. Soient la classe A définie par `class A {int i; A(int j) {i = j; }}` et les méthodes `echanger` et `echangerbis`:

```
public static void echanger (A a1, A a2, int x1, int x2){  
    A tmp1=a1; a1=a2; a2=tmp1;int tmp2=x1; x1=x2; x2=tmp2;}  
public static void echangerbis(A a1, A a2) {int tmp = a1.i; a1.i = a2.i; a2.i = tmp;}
```

soit le code :

```
A a=new A(10); A b=new A(5); int c=3; int d =4;  
echanger(a,b,c,d); System.out.print(a.i+" "+b.i+" "+c+" "+d+" ");  
echangerbis(a, b); System.out.println(a.i + " " + b.i);
```

Quel sera l'affichage?

- (a) 10 5 3 4 5 10
- (b) 5 10 4 3 5 10
- (c) 5 10 3 4 5 10
- (d) 5 10 3 4 10 5
- (e) 10 5 3 4 10 5

Section 2. Invariants et complexité

Dans la suite $\log_a(n)$ est le logarithme de n en base a , $\log(n)$ est le logarithme en base 10 de n , et $\text{Log}(n)$ le logarithme népérien de n .

- On considère la boucle suivante: `for(int i=0; i<n; i+=2) for(int j=i; j>0; 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 $\Theta(n \log(n))$
- 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(s) qui est (sont) juste(s):
 - " 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
- On considère la boucle: `int i,j; for(i=0; i<n; i+=1){j=i; do f(j-); while(j>0)}`; 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 $\Theta(n \log(n))$
- On considère la boucle suivante: `for(int i=0; i<n; i+=1) for(int j=1; j<n; j=j+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 $\Theta(n \log(n))$
- le nombre de chiffres dans la représentation en base a d'un nombre n est:
 - un $\Theta(\log(n))$
 - un $\Theta(n)$
 - un $\Theta(a^n)$
- On considère la boucle: `for(int i=0; i<n; i+=1)for(int j=n; j>0; 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 $\Theta(n \log(n))$
 - le nombre d'appels de f est un $\Theta(2^n)$
- $\log_a(n)$ est un:
 - est un $\Theta(n)$
 - est un $\Theta(\log_b(n))$ pour tout b
 - est un $\Theta(\log^2(n))$
- 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-condition:
 - $m = \max(A, B)$
 - $1 > 0$
 - $m \geq |A - B|$
 - $A > B$

9. 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 de n (en base 10)
10. 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 $\Theta(\log(n))$
 - (c) le nombre d'appel de `f` est un $\Theta(n \log(n))$
11. On considère la boucle: `int i=0; while(i<n) f(i++);` Trouver la bonne réponse:
En fonction du *nombre k de chiffres* dans la représentation de n en base 10:
- (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 $\Theta(10^k)$
12. On considère la boucle: `for(int i=0; i<n; i+=1) 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 $\Theta(10^n)$
13. On considère le morceau de programme: `int v=0;int i=1; while(i<= n){v=v+t[i]; i++;}` Choisir parmi les assertions suivantes laquelle est FAUSSE:
- (a) " $v = \sum_{j=1}^{j=i-1} t[j]$ " est un invariant de la boucle
 - (b) " $v = \sum_{j=1}^{j=n} t[j]$ " est une post-condition pour ce programme et pour la pré-condition " t est un tableau défini au mois pour les indices de 1 à n "
 - (c) " $v = \sum_{j=1}^{j=n} t[j]$ " est un invariant de la boucle"

Section 3. Autres questions

On rappelle qu'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.

1. Le tri rapide (quicksort) trie 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

2. (Trouver la ou les bonnes réponses –attention aux négations!) Une variable static déclarée dans une classe A

- (a) ne peut être définie que dans une méthode statique
- (b) ne peut pas être modifiée
- (c) ne peut pas être initialisée
- (d) est indépendante des instances des objets de A

3. On considère:

```
Interface I{void f();}  
class A implements I{public void f(){System.out.println("A");}}
```

On considère le code suivant: `I i=new A(); i.f();`
Laquelle des assertions suivantes est correcte:

- (a) le code affichera "A"
- (b) ce code ne peut être compilé

4. Le tri par sélection trie 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

5. Si toutes les branches d'un arbre parfait ont la longueur n , le nombre de feuilles est:

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

6. Pour la classe D définie comme suit:

```
class D {  
public static int x;  
public int y;  
public static travailler() {x++;}  
public D() {x++; y- -; }  
}
```

qu'affichera le code suivant?

```
D.travailler(); D a=new D(); D b=new D(); a.travailler();  
System.out.println(b.x + " et " + b.y);
```

- (a) 2 et -1
- (b) ce code ne se compile pas
- (c) 4 et -2
- (d) 4 et -1

7. 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 le tableau *trié* t entre les indices l et r . Quelle le nombre de comparaisons en fonction de $n = r - l$ (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

8. En supposant que la class A est définie par `class A {int i;}`. Parmi les morceaux de code suivant lesquels sont corrects (peuvent être compilés et exécutés):

- (a) `A a=new A(); Object o=a; A b= (A) o;`
- (b) `A a=new A(); Object o=a; A b= o;`
- (c) `Object o=new Object(); A a=(A)o;`

9. On considère le programme suivant:

```
static boolean cherche(int v,int [] t, int n)
{for(int i=0;i<=n;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 0 et n 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

10. Le tri-fusion trie 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

11. Si un arbre parfait a n feuilles, le nombre total de noeuds (attention: les feuilles sont des noeuds) est

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

12. Pour un arbre binaire parfait ayant n feuilles, toutes les branches ont la longueur:

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

Answer Key for Exam A

Section 1. Questions générales

1. Pour la classe définie comme suit:

```
public class Bidon {int x;};
```

lequel des programmes est faux (ne peut s'exécuter)?

- (a) Bidon a=new Bidon(3);
- (b) Bidon a=new Bidon(); String s=a.toString();
- (c) Bidon a=new Bidon(); boolean b= a.equals("bonjour");
- (d) Bidon a=new Bidon(); boolean b= a.equals(a);

2. On considère la classe définie par :

```
class A {int i; static void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=a; a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

3. On considère la classe définie par : class B {int i=0; }

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a,b; a.i=10; b=a; b.i=5; System.out.println(a.i);
```

- (a) il affiche 10
- (b) il affiche 5
- (c) il provoque une erreur

4. On considère la classe définie par : class B {int i; }

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B();B b=new B(); a.i=10; b=a; b.i=5;  
if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");
```

- (a) il affiche EGAL
- (b) il affiche PAS EGAL
- (c) il provoque une erreur

5. On définit la méthode permuter par:

```
public static void permuter (String s1, String s2, int x1, int x2){  
    String tmp1=s1; s1=s2; s2=tmp1;  
    int tmp2=x1; x1=x2; x2=tmp2;  
}
```

On l'applique dans le contexte suivant:

```
String a="un"; String b="deux"; int c=3; int d =4; permuter(a,b,c,d);
```

Quelles seront les valeurs de a,b,c,d après l'exécution de ce code?

- (a) "un", "deux", 3, 4
- (b) "deux", "un", 3, 4
- (c) "un", "deux", 4, 3
- (d) "deux", "un", 4, 3

6. On considère la classe définie par :

```
class A {static int i; void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

7. On considère la classe définie par : `class B {int i; }` Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B(); B b=new B(); b.i=10; b=a; b.i=5; System.out.println(a.i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

8. Pour la classe définie comme suit:

```
public class Bidon {int x=5;};
```

Parmi les programmes suivants le(s)quel(s) provoque(nt) une erreur?

- (a) Bidon a=new Bidon(3);
- (b) Bidon a=new Bidon(); String s="bidon"+a;

9. On considère la classe définie par :

```
A {static int i; static void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

10. On considère la classe définie par : `class C {static int i; }`

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
C a=new C(); C b=new C(); a.i=10; b.i=5; System.out.println(a.i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

11. Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
int i,j; i=10; j=i; j=5; System.out.println(i);
```

- (a) il affiche 5
- (b) il affiche 10
- (c) il provoque une erreur

12. Pour la classe D définie comme suit:

```
class D {  
int x;  
D() {x=3; };  
D( int a){this(); x=x+a;};  
D( int a, int b){this(b); x= x-a;}}
```

qu'affichera le code suivant?

```
D a=new D(5,6);  
System.out.println(a.x);
```

- (a) 1
- (b) 2
- (c) 3
- (d) 4

13. On considère la classe définie par : `class B {int i; }`

Quel est le résultat du morceau de code suivant (trouver la bonne réponse)

```
B a=new B(); B b=new B(); a.i=10; b.i=10;  
if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");
```

- (a) il affiche EGAL
- (b) il affiche PAS EGAL
- (c) il provoque une erreur

14. On considère la classe définie par :

```
class A {int i; void print(){System.out.println(i); }}
```

On considère le morceau de code suivant (trouver la bonne réponse)

```
A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();
```

- (a) il affiche 5
- (b) il affiche 10
- (c) la classe A ne peut être compilée

15. Soient la classe A définie par `class A {int i; A(int j) {i = j; }}` et les méthodes `echanger` et `echangerbis`:

```
public static void echanger (A a1, A a2, int x1, int x2){  
A tmp1=a1; a1=a2; a2=tmp1;int tmp2=x1; x1=x2; x2=tmp2;}  
public static void echangerbis(A a1, A a2) {int tmp = a1.i; a1.i = a2.i; a2.i = tmp;}
```

soit le code :

```
A a=new A(10); A b=new A(5); int c=3; int d =4;  
echanger(a,b,c,d); System.out.print(a.i+" "+b.i+" "+c+" "+d+" ");  
echangerbis(a, b); System.out.println(a.i + " " + b.i);
```

Quel sera l'affichage?

- (a) 10 5 3 4 5 10
- (b) 5 10 4 3 5 10
- (c) 5 10 3 4 5 10
- (d) 5 10 3 4 10 5
- (e) 10 5 3 4 10 5

Section 2. Invariants et complexité

Dans la suite $\log_a(n)$ est le logarithme de n en base a , $\log(n)$ est le logarithme en base 10 de n , et $\text{Log}(n)$ le logarithme népérien de n .

- On considère la boucle suivante: `for(int i=0; i<n; i+=2) for(int j=i; j>0; 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 $\Theta(n \log(n))$
- 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(s) qui est (sont) juste(s):
 - "`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
- On considère la boucle: `int i,j; for(i=0; i<n; i+=1){j=i; do f(j--); while(j>0)};` 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 $\Theta(n \log(n))$
- On considère la boucle suivante: `for(int i=0; i<n; i+=1) for(int j=1; j<n; j=j+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 $\Theta(n \log(n))$
- le nombre de chiffres dans la représentation en base a d'un nombre n est:
 - un $\Theta(\log(n))$
 - un $\Theta(n)$
 - un $\Theta(a^n)$
- On considère la boucle: `for(int i=0; i<n; i+=1)for(int j=n; j>0; 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 $\Theta(n \log(n))$
 - le nombre d'appels de f est un $\Theta(2^n)$
- $\log_a(n)$ est un:
 - est un $\Theta(n)$
 - est un $\Theta(\log_b(n))$ pour tout b
 - est un $\Theta(\log^2(n))$
- 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-condition:
 - $m = \max(A, B)$
 - $1 > 0$
 - $m \geq |A - B|$
 - $A > B$

9. 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 de n (en base 10)
10. 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 $\Theta(\log(n))$
 - (c) le nombre d'appel de `f` est un $\Theta(n \log(n))$
11. On considère la boucle: `int i=0; while(i<n) f(i++)`; Trouver la bonne réponse:
En fonction du *nombre k de chiffres* dans la représentation de n en base 10:
- (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 $\Theta(10^k)$
12. On considère la boucle: `for(int i=0; i<n; i+=1) 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 $\Theta(10^n)$
13. On considère le morceau de programme: `int v=0;int i=1; while(i<= n){v=v+t[i]; i++;}` Choisir parmi les assertions suivantes laquelle est FAUSSE:
- (a) " $v = \sum_{j=1}^{j=i-1} t[j]$ " est un invariant de la boucle
 - (b) " $v = \sum_{j=1}^{j=n} t[j]$ " est une post-condition pour ce programme et pour la pré-condition " t est un tableau défini au mois pour les indices de 1 à n "
 - (c) " $v = \sum_{j=1}^{j=n} t[j]$ est un invariant de la boucle"

Section 3. Autres questions

On rappelle qu'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.

1. Le tri rapide (quicksort) trie 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

2. (Trouver la ou les bonnes réponses –attention aux négations!) Une variable static déclarée dans une classe A

- (a) ne peut être définie que dans une méthode statique
- (b) ne peut pas être modifiée
- (c) ne peut pas être initialisée
- (d) est indépendante des instances des objets de A

3. On considère:

```
Interface I{void f();}  
class A implements I{public void f(){System.out.println("A");}}
```

On considère le code suivant: `I i=new A(); i.f();`

Laquelle des assertions suivantes est correcte:

- (a) le code affichera "A"
- (b) ce code ne peut être compilé

4. Le tri par sélection trie 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

5. Si toutes les branches d'un arbre parfait ont la longueur n , le nombre de feuilles est:

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

6. Pour la classe D définie comme suit:

```
class D {  
public static int x;  
public int y;  
public static travailler() {x++;}  
public D() {x++; y- -; }  
}
```

qu'affichera le code suivant?

```
D.travailler(); D a=new D(); D b=new D(); a.travailler();  
System.out.println(b.x + " et " + b.y);
```

- (a) 2 et -1
- (b) ce code ne se compile pas
- (c) 4 et -2
- (d) 4 et -1

7. 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 le tableau *trié* t entre les indices l et r . Quelle le nombre de comparaisons en fonction de $n = r - l$ (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

8. En supposant que la class A est définie par `class A {int i;}`. Parmi les morceaux de code suivant lesquels sont corrects (peuvent être compilés et exécutés):

- (a) `A a=new A(); Object o=a; A b= (A) o;`
- (b) `A a=new A(); Object o=a; A b= o;`
- (c) `Object o=new Object(); A a=(A)o;`

9. On considère le programme suivant:

```
static boolean cherche(int v,int [] t, int n)
{for(int i=0;i<=n;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 0 et n 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

10. Le tri-fusion trie 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

11. Si un arbre parfait a n feuilles, le nombre total de noeuds (attention: les feuilles sont des noeuds) est

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

12. Pour un arbre binaire parfait ayant n feuilles, toutes les branches ont la longueur:

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