

POO

QCM 1, Version:

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	14	15	16	17	18	19	20
Réponse(s)																				

Bonne réponse=1pt; mauvaise réponse ou réponse incomplète =-0,5pt; pas de réponse=0pt. (Toutes les questions ont au moins une bonne réponse). Le total est sur 17.

1. On considère la classe définie par : `class A {static int i; static void print(){System.out.println(i); }}`.
L’exécution de: `A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();`

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

2. On considère la classe définie par : `class A {int i; }`
L’exécution de: `A a=new A(); A b=new A(); a.i=8; b=a; b.i=5; System.out.println(a.i);`

- (a) affiche 5
- (b) affiche 8
- (c) provoque une erreur

3. `class A{
 static int i=0;int j=10;
 static void g(){System.out.println(j);}
}`

avec le morceau de code : `(new A()).g();`

- (a) provoque une erreur à la compilation
- (b) affiche 10
- (c) provoque une erreur à l’exécution

4. Soit: `class A{private int i=10; public int geti(){return i;}}`
Supposons que l’on est à l’intérieur d’une méthode f qui n’est pas définie dans la classe A et que a est un objet de la classe A: (choisir la ou les bonnes réponses)

- (a) après: `int i; i=a.i;` un test: `i==10` retourne `true`
- (b) après: `int i; i=a.geti();` un test: `i==10` retourne `true`

5. On considère la classe :

```
class C{int v; C(int i){v=i;}public String toString(){return v+" "};}
```

Et la méthode: `public static void p1(C x, C y){int tmp=x.v; x.v=y.v;y.v=tmp; }`

L'exécution du morceau de code :

`C c1= new C(1); C c2 = new C(2);p1(c1, c2); System.out.println(c1+" "+c2);` affichera:

- (a) 1 2
- (b) 2 1
- (c) provoque une erreur à la compilation

6. On considère les classes : `class A {int i=0; B b; }`, et: `class B {int j=0;}` ainsi que le code `A a1=new A(); A a2=new A(); B b=new B();`

Quelle(s) expression(s) ont la valeur true:

- (a) `(a1.i != b.j)`
- (b) `(a1.b == a2.b)`
- (c) `(a1 == a2)`

7. On considère la classe :

```
class D{int v;
    D(int i){v=i;}
    public String toString(){return v+" "};
    void ech(D d){int tmp=this.v; this.v = d.v;d.v=tmp;}
}
```

L'exécution du morceau de code :

`D d1=new D(1); D d2= new D(-1); d1.ech(d2); System.out.println(d1+" "+d2);` affichera:

- (a) 1 -1
- (b) -1 1
- (c) provoque une erreur à la compilation

8. On considère la classe définie par : `class A {int i; static void print(){System.out.println(i); }}`. L'exécution de: `A a=new A(); A b=a; a.i=5; b.i=10; a.print();`

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

9. On considère la classe définie par : `class A {int i; }`. L'exécution de:

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

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

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

10. On considère la classe définie par : `class A {int i=0; }`. L'exécution de:

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

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

11. class A{
 static int i=0;int j=10;
 void f(){System.out.println("i="+i);System.out.println(" j="+j);}
 static void g(){this.f();}
 }

avec le morceau de code: (new A()).g();

- (a) provoque une erreur à la compilation
- (b) affiche: i=0 j=10
- (c) provoque une erreur à l'exécution

12. On considère la classe définie par : class A {int i; void print(){System.out.println(i); }}
 L'exécution de A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();

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

13. Pour la classe D définie comme suit:

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

qu'affichera le code : D a=new D(5,6); System.out.println(a.x);

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

14. On considère la classe définie par : class A {int i; }. L'exécution de:
 A a=new A();A b=new A(); a.i=10; b=a; b.i=5;
 if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");

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

15. Soit: class A{private int i=10; public int geti(){return i;}}
 Pour le code suivant: public static boolean egal(A a, A b){return a.i ==b.i; } défini dans une classe
autre que A, un appel egal(new A(),new A()):

- (a) retourne true
- (b) provoque une erreur à la compilation

16. On considère la classe définie par : class A {static int i; }. L'exécution de:
 A a=new A(); A b=new A(); a.i=10; b.i=5; System.out.println(a.i);

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

17. On considère la classe définie par : `class A {static int i; void print(){System.out.println(i); }}`
L'exécution de `A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();`

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

18. On considère la classe :

```
class C{int v; C(int i){v=i;}public String toString(){return v+" "};}
```

Et la méthode: `public static void p2(C x, C y){C tmp = x; x = y ; y = tmp; }` L'exécution du morceau de code :

`C c1= new C(1); C c2 = new C(2);p1(c1, c2); System.out.println(c1+" "+c2);` affichera:

- (a) 1 2
- (b) 2 1
- (c) provoque une erreur à la compilation

19.

```
class A{
    static int i=0;int j=10;
    void f(){System.out.println("i="+i);System.out.println(" j="+j);}
}
```

avec le morceau de code: `(new A()).f();`

- (a) provoque une erreur à la compilation
- (b) affiche i=0 j=10
- (c) provoque une erreur à l'exécution

20. On définit la méthode `permuter`:

```
public static void permuter(String s1,String s2,int i, int j){
    String ts=s1; int ti=i; s1=s2; s2=ts;i=j;j=ti;}
}
```

Considérons le morceau de code suivant:

`String s="au "; String t="revoir ";int i=5; int j=8; permuter(s,t,i,j); System.out.println(s+t+i+" "+j);`
affichera ?

- (a) au revoir 5 8
- (b) revoir au 5 8
- (c) au revoir 8 5
- (d) revoir au 8 5

Answer Key for Exam A

Bonne réponse=1pt; mauvaise réponse ou réponse incomplète =-0,5pt; pas de réponse=0pt. (Toutes les questions ont au moins une bonne réponse). Le total est sur 17.

1. On considère la classe définie par : `class A {static int i; static void print(){System.out.println(i); }}`.
L'exécution de: `A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();`

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

2. On considère la classe définie par : `class A {int i; }`
L'exécution de: `A a=new A(); A b=new A(); a.i=8; b=a; b.i=5; System.out.println(a.i);`

- (a) affiche 5
- (b) affiche 8
- (c) provoque une erreur

3. `class A{
 static int i=0;int j=10;
 static void g(){System.out.println(j);}
}`

avec le morceau de code : `(new A()).g();`

- (a) provoque une erreur à la compilation
- (b) affiche 10
- (c) provoque une erreur à l'exécution

4. Soit: `class A{private int i=10; public int geti(){return i;}}`
Supposons que l'on est à l'intérieur d'une méthode f qui n'est pas définie dans la classe A et que a est un objet de la classe A: (choisir la ou les bonnes réponses)

- (a) après: `int i; i=a.i;` un test: `i==10` retourne true
- (b) après: `int i; i=a.geti();` un test: `i==10` retourne true

5. On considère la classe :

```
class C{int v; C(int i){v=i;}public String toString(){return v+" "}}
```

Et la méthode: `public static void p1(C x, C y){int tmp=x.v; x.v=y.v;y.v=tmp; }`

L'exécution du morceau de code :

`C c1= new C(1); C c2 = new C(2);p1(c1, c2); System.out.println(c1+" "+c2);` affichera:

- (a) 1 2
- (b) 2 1
- (c) provoque une erreur à la compilation

6. On considère les classes : `class A {int i=0; B b; }, et: class B {int j=0;}` ainsi que le code `A a1=new A(); A a2=new A(); B b=new B();`
Quelle(s) expression(s) ont la valeur true:

- (a) `(a1.i != b.j)`
- (b) `(a1.b == a2.b)`
- (c) `(a1 == a2)`

7. On considère la classe :

```
class D{int v;  
    D(int i){v=i;}  
    public String toString(){return v+" "};  
    void ech(D d){int tmp=this.v; this.v = d.v;d.v=tmp;}  
}
```

L'exécution du morceau de code :

D d1=new D(1); D d2= new D(-1); d1.ech(d2); System.out.println(d1+" "+d2); affichera:

- (a) 1 -1
- (b) -1 1
- (c) provoque une erreur à la compilation

8. On considère la classe définie par : class A {int i; static void print(){System.out.println(i); }}. L'exécution de: A a=new A(); A b=a; a.i=5; b.i=10; a.print();

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

9. On considère la classe définie par : class A {int i; }. L'exécution de:

A a=new A(); A b=new A(); a.i=10; b.i=10;

if(a==b)System.out.println("EGAL"); else System.out.println("PAS EGAL");

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

10. On considère la classe définie par : class A {int i=0; }. L'exécution de:

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

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

11. class A{

```
    static int i=0;int j=10;  
    void f(){System.out.println("i="+i);System.out.println(" j="+j);}  
    static void g(){this.f();}  
}
```

avec le morceau de code: (new A()).g();

- (a) provoque une erreur à la compilation
- (b) affiche: i=0 j=10
- (c) provoque une erreur à l'exécution

12. On considère la classe définie par : class A {int i; void print(){System.out.println(i); }}

L'exécution de A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();

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

13. Pour la classe D définie comme suit:

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

qu'affichera le code : `D a=new D(5,6); System.out.println(a.x);`

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

14. On considère la classe définie par : `class A {int i; }`. L'exécution de:

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

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

15. Soit: `class A{private int i=10; public int geti(){return i;}}`

Pour le code suivant: `public static boolean egal(A a, A b){return a.i ==b.i; }` défini dans une classe *autre* que A, un appel `egal(new A(),new A());`:

- (a) retourne true
- (b) provoque une erreur à la compilation

16. On considère la classe définie par : `class A {static int i; }`. L'exécution de:

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

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

17. On considère la classe définie par : `class A {static int i; void print(){System.out.println(i); }}`

L'exécution de `A a=new A(); A b=new A(); a.i=5; b.i=10; a.print();`

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

18. On considère la classe :

```
class C{int v; C(int i){v=i;}public String toString(){return v+" "};}
```

Et la méthode: `public static void p2(C x, C y){C tmp = x; x = y ; y = tmp; }` L'exécution du morceau de code :

```
C c1= new C(1); C c2 = new C(2);p1(c1, c2); System.out.println(c1+" "+c2);
```

 affichera:

- (a) 1 2
- (b) 2 1
- (c) provoque une erreur à la compilation

19. class A{
 static int i=0;int j=10;
 void f(){System.out.println("i="+i);System.out.println(" j="+j);}
}

avec le morceau de code: (new A()).f();

- (a) provoque une erreur à la compilation
- (b) affiche i=0 j=10
- (c) provoque une erreur à l'exécution

20. On définit la méthode permuter:

```
public static void permuter(String s1,String s2,int i, int j){  
    String ts=s1; int ti=i; s1=s2; s2=ts;i=j;j=ti;}  
}
```

Considérons le morceau de code suivant:

String s="au "; String t="revoir ";int i=5; int j=8; permuter(s,t,i,j); System.out.println(s+t+i+" "+j);
affichera ?

- (a) au revoir 5 8
- (b) revoir au 5 8
- (c) au revoir 8 5
- (d) revoir au 8 5