

NOM :

---

PRENOM :

---

## Cours C# - POO - Interrogation 1 - 23/02/2021

---

### I - Les Pointeurs

---

1) Une variable de type pointeur a en memoire la valeur de la variable pointée.

Vrai ou Faux ?

Réponse :

2) '&' devant une variable permet d'obtenir la valeur de cette variable.

Vrai ou Faux ?

Réponse :

3) L'opérateur '\*' devant une variable de type pointeur permet d'accéder à la valeur de la variable pointée par ce pointeur.

Vrai ou Faux ?

Réponse :

4) L'opérateur qui permet d'accéder à la propriété d'une structure via un pointeur est '\*'.

Vrai ou Faux ?

Réponse :

5) Soit le code suivant :

```
int* monPointeur;
```

Quelle est la valeur de la variable **monPointeur** ?

Réponse :

6) Soit une variable `int* aPointer` qui pointe vers une variable à l'adresse `x1221458214`.

Si j'utilise l'opérateur '+' une fois sur le pointeur `aPointer`, à quelle adresse pointera-t-il ?

Réponse :

### II - Les Classes

---

#### Instanciación

---

1) On ne peut instancier qu'un objet à partir d'une classe.

Vrai ou Faux ?

Réponse :

2) Une classe décrit un type de données.

Vrai ou Faux ?

Réponse :

3) Une classe permet de créer plusieurs objets de même caractéristiques.

Vrai ou Faux ?

Réponse :

4) Une classe peut porter plusieurs noms différents.

Vrai ou Faux ?

Réponse :

#### Membres d'instance

---

1) Tous les objets d'une même classe possèdent les mêmes propriétés et les mêmes méthodes d'instance.

Vrai ou Faux ?

Réponse :

2) Toutes les propriétés des objets d'une même classe possèdent des valeurs identiques.

Vrai ou Faux ?

Réponse :

3) Les propriétés et les méthodes d'instance font partis des membres d'instance d'une classe.

Vrai ou Faux ?

Réponse :

4) Une classe doit contenir des propriétés d'instance.

Vrai ou Faux ?

Réponse :

5) Une classe doit contenir des méthodes d'instance.

Vrai ou Faux ?

Réponse :

## Référence et création d'instances

---

1) Si je crée trois références alors elles désignent trois objets différents.

Vrai ou Faux ?

**Réponse :**

2) Il peut exister plusieurs variables qui référencent un même objet.

Vrai ou Faux ?

**Réponse :**

3) Il existe trois catégories de variables : valeur, référence et objet.

Vrai ou Faux ?

**Réponse :**

4) La déclaration de variable

```
A v;
```

,où A est un identificateur de classe, ne provoque pas la création d'un objet de type A.

Vrai ou Faux ?

**Réponse :**

5) Pour créer un nouvel objet à partir d'un objet existant, je déclare une variable que j'initialise avec une référence vers cet objet.

Vrai ou Faux ?

**Réponse :**

6) L'instruction suivante est-elle valide ?

```
Voiture v = v;
```

Oui ou Non ?

**Réponse :**

7) Les instructions suivantes sont-elles valides ?

```
Voiture v = new Voiture();  
v = v;
```

Oui ou Non ?

**Réponse :**

## Accessibilité et Visibilité

---

1) Les membres d'une classe dont la visibilité n'est pas précisée sont considérés comme privés.

Vrai ou Faux ?

**Réponse :**

2) Les seuls membres privés auxquels je peux accéder sont ceux de l'instance courante this.

Vrai ou Faux ?

**Réponse :**

3) Une classe B peut accéder aux membres privés d'une classe A si elles sont déclarées dans le même fichier.

Vrai ou Faux ?

**Réponse :**

4) Dans une même classe, je ne peux pas accéder à un membre privé depuis un membre public.

Vrai ou Faux ?

**Réponse :**

5) Les règles de visibilité sont vérifiées à la compilation.

Vrai ou Faux ?

**Réponse :**

## Constructeur

---

1) Une classe a toujours un constructeur.

Vrai ou Faux ?

**Réponse :**

2) Un constructeur peut avoir un type de retour s'il s'agit d'une référence.

Vrai ou Faux ?

**Réponse :**

3) L'opérateur return ne peut pas être utilisé dans un constructeur.

Vrai ou Faux ?

**Réponse :**

4) Lors de l'appel à l'opérateur new, le constructeur sans argument est toujours appelé.

Vrai ou Faux ?

**Réponse :**

5) La référence vers l'objet courant `this` n'existe pas dans un constructeur.

Vrai ou Faux ?

**Réponse :**

6) Un constructeur doit avoir le même nom que la classe où il est déclaré.

Vrai ou Faux ?

**Réponse :**

7) On est obligé d'initialiser la valeur des champs dans un constructeur.

Vrai ou Faux ?

**Réponse :**

8) Une classe peut avoir plusieurs constructeurs.

Vrai ou Faux ?

**Réponse :**

## Opérateur d'accès

---

1) Si je change la valeur d'un champ d'instance, l'objet n'est plus considéré comme appartenant à sa classe d'origine.

Vrai ou Faux ?

**Réponse :**

2) Une fois l'objet créé et ses variables d'instance initialisées, je ne peux plus les modifier.

Vrai ou Faux ?

**Réponse :**

3) Je peux accéder directement à un objet sans passer par une référence.

Vrai ou Faux ?

**Réponse :**

4) Si j'appelle une méthode depuis une référence et que cette référence a pour valeur `null` alors le programme plante.

Vrai ou Faux ?

**Réponse :**

5) Soient deux références `r1` et `r2` qui désignent le même objet, si je change `r1` pour qu'elle réfère un autre objet, alors `r2` désigne aussi ce nouvel objet.

Vrai ou Faux ?

**Réponse :**

6) Si aucune référence ne désigne un objet, alors je ne peux pas accéder aux attributs de cet objet.

Vrai ou Faux ?

**Réponse :**

7) Dans une méthode d'instance, `this` est toujours une référence vers un objet de la classe courante.

Vrai ou Faux ?

**Réponse :**

8) Dans une méthode d'instance, `this.v` et `v` peuvent désigner 2 choses différentes.

Vrai ou Faux ?

**Réponse :**

## III - L'Héritage

---

1) Une classe peut avoir plusieurs classes mères.

Vrai ou Faux ?

**Réponse :**

2) Une classe peut avoir plusieurs ancêtres.

Vrai ou Faux ?

**Réponse :**

3) Une classe peut avoir plusieurs classes filles.

Vrai ou Faux ?

**Réponse :**

4) Une classe peut avoir plusieurs descendants.

Vrai ou Faux ?

**Réponse :**

5) Plusieurs classes peuvent hériter d'une même classe.

Vrai ou Faux ?

**Réponse :**

## Visibilité et héritage

---

```
class A
{
    public int attributPublicA;
    protected int attributProtectedA;
    private int attributPrivateA;
}

class B : A
{}

class C
{}

```

1) La classe B hérite de attributPublicA

Vrai ou Faux ?

**Réponse :**

2) Dans la classe B on peut accéder à attributPublicA depuis l'instance courante.

Vrai ou Faux ?

**Réponse :**

3) La classe B hérite de attributProtectedA

Vrai ou Faux ?

**Réponse :**

4) Dans la classe B on peut accéder à attributProtectedA depuis l'instance courante.

Vrai ou Faux ?

**Réponse :**

5) La classe B hérite de attributPrivateA

Vrai ou Faux ?

**Réponse :**

6) Dans la classe B on peut accéder à attributPrivateA depuis l'instance courante.

Vrai ou Faux ?

**Réponse :**

7) Dans la classe C on peut accéder à attributPublicA depuis un référence sur un objet de type A.

Vrai ou Faux ?

**Réponse :**

8) Dans la classe C on peut accéder à attributProtectedA depuis un référence sur un objet de type A.

Vrai ou Faux ?

**Réponse :**

9) Dans la classe C on peut accéder à attributPrivateA depuis un référence sur un objet de type A.

Vrai ou Faux ?

**Réponse :**

10) Dans la classe C on peut accéder à attributPublicA depuis un référence sur un objet de type B.

Vrai ou Faux ?

**Réponse :**

11) Dans la classe C on peut accéder à attributProtectedA depuis un référence sur un objet de type B.

Vrai ou Faux ?

**Réponse :**

12) Dans la classe C on peut accéder à attributPrivateA depuis un référence sur un objet de type B.

Vrai ou Faux ?

**Réponse :**

# Type de varibale et Héritage

---

```
class Mere
{
    public int attributMere;
    public void methodeMere(){};
}

class Fille : Mere
{
    public int attributFille;
    public void methodeFille(){};
}

...
Mere a = new Mere();
Fille b = new Fille();
Mere c = new Fille();
```

Quel est le type déclaré de **a** ?

**Réponse :**

Quel est le type déclaré de **b** ?

**Réponse :**

Quel est le type déclaré de **c** ?

**Réponse :**

Quel est le type réel de **a** ?

**Réponse :**

Quel est le type réel de **b** ?

**Réponse :**

Quel est le type réel de **c** ?

**Réponse :**

Avec **a** j'ai accès à **attributMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **b** j'ai accès à **attributMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **c** j'ai accès à **attributMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **a** j'ai accès à **MethodeMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **b** j'ai accès à **MethodeMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **c** j'ai accès à **MethodeMere** ?

Vrai ou Faux ?

**Réponse :**

Avec **a** j'ai accès à **attributFille** ?

Vrai ou Faux ?

**Réponse :**

Avec **b** j'ai accès à **attributFille** ?

Vrai ou Faux ?

**Réponse :**

Avec **c** j'ai accès à **attributFille** ?

Vrai ou Faux ?

**Réponse :**

Avec **a** j'ai accès à **MethodeFille** ?

Vrai ou Faux ?

**Réponse :**

Avec **b** j'ai accès à **MethodeFille** ?

Vrai ou Faux ?

**Réponse :**

Avec **c** j'ai accès à **MethodeFille** ?

Vrai ou Faux ?

**Réponse :**