



Durée : 1h30

Documents autorisés : Polycopié de cours, TDs et TP.

Exercice 1 : Membres de classe (4 points)

- Qu'est-ce qu'un membre de classe et comment peut-on y accéder ?
- Les membres d'instance ont-ils accès aux membres de classe ?
- Les membres de classe ont-ils accès aux membres d'instance ?
- Faites le lien entre la notion de membres de classe et la notion de métaclasse.

Exercice 2 : Accessibilité (4 points)

- Quels sont les différents modificateurs d'accessibilité (en Java et C++) et à quoi donnent-ils accès ?
- Quels modificateurs applique-t-on habituellement aux différents membres d'une classe (en Java et C++) et pourquoi ?
- Quels sont les modes par défaut (en Java et C++) lorsque l'on ne précise aucun modificateur d'accès ?
- Ces modificateurs sont-ils conservés (en Java et C++) lors de l'héritage ? Et peuvent-ils être modifiés lors de l'héritage (en Java et C++) ?

Exercice 3 : Appels implicites en C++ (5 points)

En étudiant le morceau de code C++ ci-dessous, déterminez l'affichage déclenché par chacune des lignes de code du programme principal (Programme.cpp, page suivante). Un exemple d'affichage est donné pour la première ligne de code

A.h (définition de la classe A)	A.cpp (implémentation de la classe A)
<pre>class A { public : // Constructeur par défaut A(void); // constructeur de copie par // référence A(const A & a); // constructeur valué A(int val); // lecture de la valeur int getVal(void) const; // écriture de la valeur void setVal(int val); // lecture du numéro d'objet. int getNumero() const; protected : int val; int numero; static int compteur; }; // réimplémentation de l'opérateur // de sortie standard pour le type A ostream & operator << (ostream & out, const A & a);</pre>	<pre>// initialisation des numéros int A::compteur = 0; // Constructeur par défaut A::A(void) : val(0), numero(++compteur) { cout << "Constructeur par défaut() : " << *this << endl; } // constructeur de copie par référence A::A(const A & a) : val(a.val), numero(++compteur) { cout << "Constructeur de copie par référence(" << a << ") : " << *this << endl; } // constructeur valué A::A(int v) : val(v), numero(++compteur) { cout << "Constructeur valué(" << val << ") : " << *this << endl; } // lecture de la valeur int A::getVal(void) const { return val; }</pre>

A.cpp (implémentation de la classe A, suite)

```
// écriture de la valeur
void A::setVal(int val)
{
    this->val = val;
}
// lecture du numéro d'objet.
int A::getNumero() const
{
    return numero;
}
// réimplémentation de l'opérateur de sortie standard pour le type A
ostream & operator << (ostream & out, const A & a)
{
    out << "A[" << a.getNumero() << "]" ("
        << a.getVal() << ")";
    return out;
}
```

Programme.cpp (Utilisation de la classe A)

```
A creerObjet(int val)
{
    cout << "Création d'objet ("
        << val << ")" << endl;
    return A(val);
}
```

```
A copierObjet(A a)
{
    cout << "Copie de l'objet ("
        << a << ")" << endl;
    A a2 = a;
    return a2;
}
```

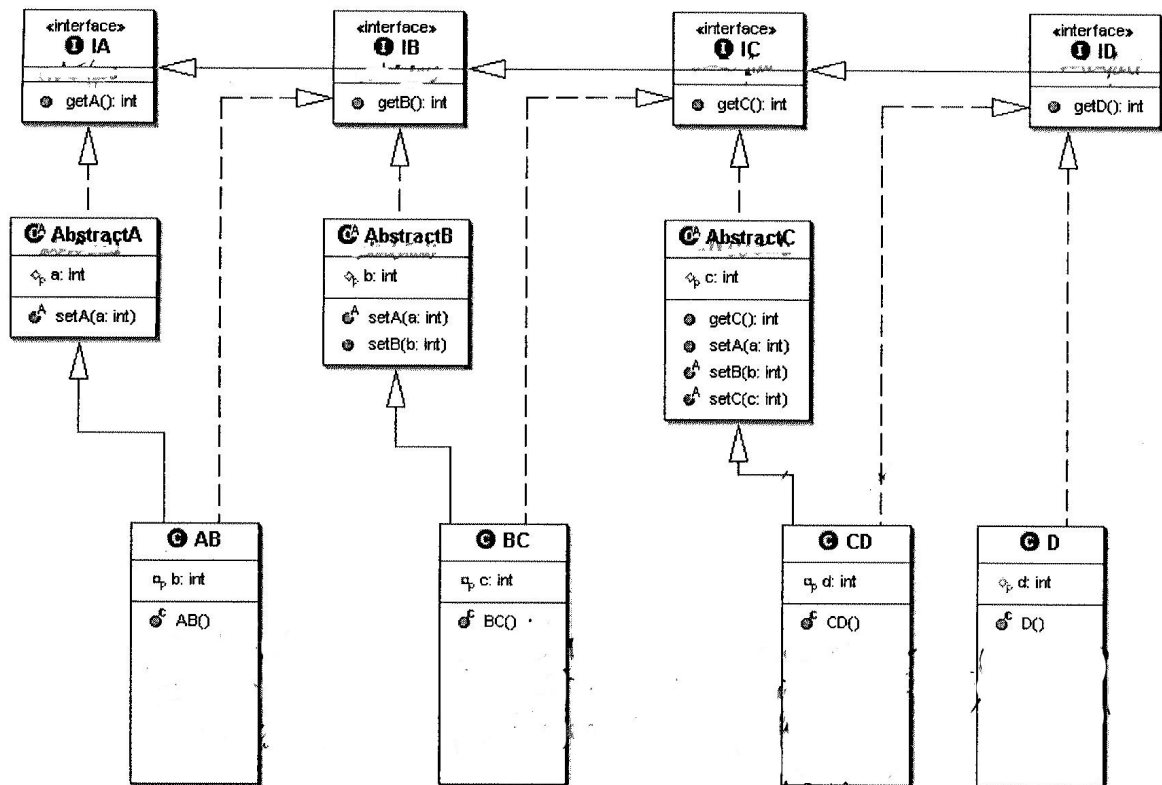
```
int main(void)
{
```

	Code	Affichage
1	A a1(2);	Constructeur valué(2) : A[1](2)
2	A a2 = a1;	
3	A a3 = creerObjet(3);	
4	A a4;	
5	cout << "Objet a4 = " << a4 << endl;	
6	a4 = copierObjet(a3);	

```
return 0;
}
```

Exercice 4 : Héritage (8 points)

Soit la hiérarchie de classes JAVA suivante :



Légende :

- Les interfaces sont indiquées par : **I**
- Les classes sont indiquées par : **C**
- Les membres publics sont précédés de : **+**
- Les membres protégés sont précédés de : **+**
- Les membres privés sont précédés de : **-**
- Les relations d'héritage sont indiquées par les flèches en traits pleins.
- Les relations d'implémentation sont indiquées par les flèches en traits pointillés.
- Les membres ou les classes abstraite(s) sont indiqués par : **A**
 - o Pour une classe abstraite, on aura donc : **C^A**
 - o Pour une méthode publique et abstraite, on aura donc : **+^A**

Complétez les classes AB, BC, CD & D avec les méthodes qui doivent être implémentées (et seulement celles-ci) dans ces classes.

Pour cet exercice, vous pourrez répondre directement sur cette feuille en y indiquant toutefois votre numéro de copie.