

Interrogation

Nom : Prénoms : Groupe :

Exercice 1.

1. Qu'est ce qui définit le contrat d'une classe ?

.....

.....

2. Dans quel cas aurons nous besoin d'utiliser :

- une classe abstraite
- une méthode finale
- une variable de classe
- la redéfinition

3. Quelle est la différence entre une exception contrôlée et une exception non contrôlée?

.....

.....

Exercice 2.

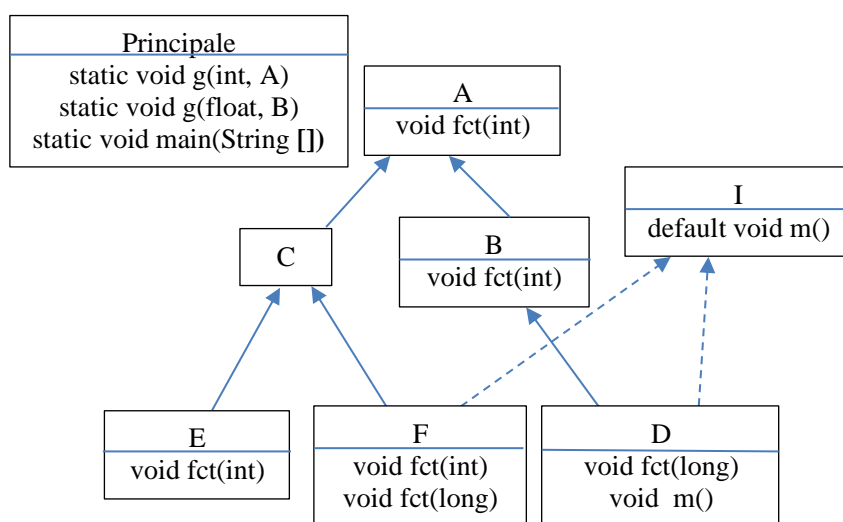
Soit la hiérarchie de classes illustrée dans la figure ci-contre:

A, B, C, D, E et F sont des classes, alors que I est une interface implémentée par les classes D et F.

Les signatures des méthodes sont illustrées sur le diagramme.

A/ Dans la suite d'instruction suivante (tiré du *main* de la classe *Principale*), indiquez pour chacune d'elles :

- 1) si elle est correcte ou si elle provoque une erreur à la compilation.
- 2) si on peut résoudre le problème de la compilation, et si oui, s'il reste une erreur à l'exécution ou non.



Instruction	Compilation : <i>OK ou Erreur avec explication</i>	Correction en cas <i>d'erreur</i>	Exécution : <i>OK ou Erreur avec explication</i>
A o1 = new A();			
I o2 = new D();			
I o3 = new B();			
I o4 = new F();			
C o5 = new F();			
o1 = o5;			
o5=o1 ;			
B o6 = new B();			
o5 = o6;			
D o7= o6;			

B/ Nous ajoutons les instructions suivantes, en considérant les initialisations précédentes, ainsi que les variables suivantes :

int n=3; *long* y=5; *float* x=6;

Indiquez parmi ces instructions celles qui provoquent une erreur à la compilation et précisez pour les instructions correctes la méthode qui sera appelée.

Instruction	OK/méthode appelée ou Erreur avec explication
o1.fct(n);	
o2.m();	
o4.m() ;	
((D)o2).fct(n);	
o5.fct(n);	
o5.fct(y);	
g(n, o1);	
g(n, o6);	
g(x, o1);	

Corrigé

Exercice 1. 4pts

1. Qu'est ce qui définit le contrat d'une classe ? **1pt**

Les signatures/ l'ensemble de ses méthodes publiques

2. Dans quel cas aurons-nous besoin d'utiliser : **4*0,5=2pts**

- une classe abstraite **0,5**

Pour imposer de donner un corps à certaines méthodes abstraites et/ou pour factoriser le code commun entre un ensemble de classes dans une classe mère qui n'est pas instanciable

- une méthode finale **0,5**

Quand nous ne souhaitons plus que cette méthode soit redéfinie.

- une variable de classe **0,5**

Lorsqu'on a besoin d'une variable globale commune entre toutes les instances de la classes (une variable indépendante des instances).

- la redéfinition **0,5**

Lorsqu'on veut modifier le comportement d'une méthode définie dans la classe mère (sans modifier sa signature).

3. Quelle est la différence entre une exception contrôlée et une exception non contrôlée? **1pt**

Une exception contrôlée est une exception que le programme doit traiter, sinon le compilateur génère une erreur de compilation, tandis qu'une exception non contrôlée, le programme n'est pas obligé de la traiter.

Exercice 2. 6pts

A/14*0,25=3,5pts

Instruction	Compilation : <i>OK ou Erreur avec explication</i>	Correction <i>si possible</i> <i>en cas d'erreur</i>	Exécution : <i>OK ou Erreur avec explication</i>
A o1 = new A();	ok 0,25		
I o2 = new D();	ok 0,25		
I o3 = new B();	Erreur, B n'implémente pas I 0,25		
I o4 = new F();	ok 0,25		
C o5 = new F();	ok 0,25		

o1 = o5;	ok 0,25		
o5 = o1;	ERREUR, pas toute objet de classe fille est un objet de la classe mère 0,25	o5=(C)o1 0,25	Ok (o1 pointe vers o5) 0,25
B o6 = new B();	ok 0,25		
o5 = o6;	Erreur, pas de lien entre o5 et o6 0,25	<i>Pas de cast Possible</i>	
D o7= o6;	ERREUR, pas toute objet de classe fille est un objet de la classe mère 0,25	D o7= (D)o6; 0,25	Erreur O6 ne référence pas un objet de type D. 0,25

B/ (8*0,25)+0,5=2,5pts

Instruction	OK/méthode appelée ou <i>Erreur</i> <u>avec explication</u>
o1.fct(n);	Appel de fct(n) de F (o1 pointe vers un objet de type F qui redéfinit fct(int))
o2.m();	Appel de m() de D (redéfinition de default m() de I)
o4.m()	Appel de m() de I
((D)o2).fct(n);	Appel de fct(n) de B (application de la ligature dynamique) 0,5pts
o5.fct(n);	Appel de fct(n) de F (o5 pointe vers un objet de type F qui redéfinit fct(int))
o5.fct(y);	Erreur de compilation, fct(long) n'est pas connu par o5 déclaré en tant qu'objet de type C
g(n, o1);	Appel de g(int, A) de la classe Principale
g(n, o6);	Erreur de compilation, ambiguïté, les deux méthodes g conviennent
g(x, o1);	Erreur de compilation, aucune des deux méthodes g ne convient