

IPST-CNAM  
Programmation JAVA  
NFA 032  
Mercredi 25 Septembre 2013

Avec document  
Durée : **2 h30**  
Enseignant : LAFORGUE Jacques

2<sup>ème</sup> Session NFA 032

*L'examen se déroule en deux parties. Une première partie de 1h15mn, sans document, consacrée à des questions de cours, et une deuxième partie de 1h15mn, avec document, consacrée en la réalisation de programmes Java.*

*Au bout de 1h15mn, les copies de la première partie seront ramassées avant de commencer la deuxième partie.*

*Pour la première partie, vous devez rendre le QCM rempli et les réponses aux questions libres écrites sur des copies vierges.*

*Pour la deuxième partie, vous écrivez vos programmes sur des copies vierges. Vous devez écrire les codes commentés en Java.*

---

## 1<sup>ère</sup> PARTIE : COURS (sans document) Durée 1h15

---

### 1. QCM (35 points)

Mode d'emploi :

Ce sujet est un QCM dont les questions sont de 3 natures :

- **les questions à 2 propositions:** dans ce cas une seule des 2 propositions est bonne.
  - +1 pour la réponse bonne
  - -1 pour la réponse fausse
- **les questions à 3 propositions** dont 1 seule proposition est bonne
  - + 1 pour la réponse bonne
  - -1/2 pour chaque réponse fausse
- **les questions à 3 propositions** dont 1 seule proposition est fausse
  - + 1/2 pour chaque réponse bonne
  - -1 pour la réponse fausse

Il s'agit de faire une croix dans les cases de droite en face des propositions.

On peut remarquer que cocher toutes les propositions d'une question revient à ne rien cocher du tout (égal à 0).

Si vous devez raturer une croix, faites-le correctement afin qu'il n'y ait aucune ambiguïté.

N'oubliez pas d'inscrire en en-tête du QCM, votre nom et prénom.

Vous avez droit à **4 points** négatifs sans pénalité.

NOM:	PRENOM:
------	---------

En Java, la classe Hashtable<K,V> permet de gérer une collection d'éléments dont l'accès se fait par une donnée, appelée Key, et non par un rang comme cela est le cas dans la classe ArrayList		Q 1
1	OUI	
2	NON	

En Java, la classe ArrayList permet de trier les éléments même si ces éléments sont de type différent.		Q 2
1	OUI	
2	NON	

En Java, il est possible de faire l'héritage entre deux classes abstraites		Q 3
1	OUI	
2	NON	

Une classe abstraite est une classe dont les types des attributs ne sont pas connus. C'est lors de l'instanciation de l'objet que les types des attributs sont résolus		Q 4
1	OUI	
2	NON	

Soit le code suivant :		Q 5
<pre>public class A extends B implements C {     private int attr_A;     public A() {         attr_A = 10; attr_B = "TOTO"; C.attr_C = 100; } }</pre>		
1	attr_B est un attribut protected de B	
2	attr_C est un attribut static public de l'interface C	
3	attr_B est un attribut privé de B	

Soit deux classes B et C qui héritent d'une classe abstraite A. Les classes B et C peuvent utiliser par héritage les attributs protected de la classe A.		Q 6
1	OUI	
2	NON	

Une classe qui contient au moins une méthode abstraite doit être déclarée abstraite.		Q 7
1	OUI	
2	NON	

En Java, on déclare un tableau qui contient des éléments dont la classe d'appartenance implémente une interface I		Q 8
1	Cela n'est pas possible	
2	Cela est possible et on ajoute dans le tableau des objets de type I ( t[i]=new I(); )	
3	Cela est possible et on ajoute dans le tableau des objets de type B ( t[i]=new B(); ) et dont la classe B implémente l'interface I	

En JAVA, une classe peut implémenter plusieurs interfaces		Q 9
1	OUI	
2	NON	

En JAVA, une interface peut contenir des attributs :		Q 10
1	statics	
2	non statics	
3	privés	

En JAVA, une interface permet de :		Q 11
1	passer en paramètre d'une méthode un objet dont la classe d'appartenance implémente l'interface	
2	définir des traitements génériques sur des collections polymorphes	
3	créer des classes abstraites	

La déclaration d'une méthode suivante : <pre>public void traitement(String s) throws MyException</pre> précise que la méthode doit traiter l'exception MyException dans le corps de sa méthode.		Q 12
1	OUI	
2	NON	

Soit le code suivant qui ajoute un Individu dans un tableau : <pre>public void ajouter(String nom) throws Exception {     Individu ind=null;     try {         ind = rechercher(nom);     }     catch(NonTrouveException ex) {         tab[n++] = ind; } }</pre> La méthode rechercher retourne l'exception NonTrouveException si le nom de l'individu n'est pas trouvé.		Q 13
1	si l'individu que l'on veut ajouter n'est pas trouvé alors la méthode retourne l'exception NonTrouveException	
2	si l'individu que l'on veut ajouter n'est pas trouvé et si le tableau n'est pas plein alors l'individu est ajouté au tableau	
3	Si l'individu que l'on veut ajouter n'est pas trouvé et le tableau est plein alors la méthode retourne une exception prédéfinie Java (IndexOutOfBoundsException)	

Soit le code suivant :		Q 14
<pre> try{     System.out.println("AAA");     call();     System.out.println("BBB"); } catch(MyException ex) {     System.out.println("DDD"); } catch(Exception ex) {     System.out.println("CCC"); } </pre>		
avec la méthode <i>call</i> qui déclenche l'exception <i>MyException</i> .		
Ce code affiche :		
1	AAA BBB	
2	AAA DDD CCC	
3	AAA DDD	

En JAVA, un package est un regroupement de classe Java (.java et .class). Ce regroupement (création du package) se fait avec la commande de compilation (option -d) :		Q 15
<b>javac -d &lt;repertoire de regroupement&gt; *.java</b>		
1	OUI	
2	NON	

Soit une collection "liste" définie par la classe ArrayList<Individu>. Nous voulons trier les éléments de cette liste suivant 3 critères de tri différents. Pour cela :		Q 16
1	la classe Individu implémente 3 fois l'interface Comparable	
2	pour chacun des tris faire les appels : Collections.sort(liste, comparator1) Collections.sort(liste, comparator2) ou Collections.sort(liste, comparator3) où comparator1, comparator2, comparator3 sont des instances des classes Comparator1, Comparator2, Comparator3 qui implémentent la méthode compare de l'interface Comparator<Individu>	
3	la classe Individu implémente l'interface Comparable et on code dans la méthode compareTo les 3 critères de tri. Le choix du tri se fait par la valeur d'un attribut statique	

Soit deux thread t1 et t2 qui utilisent la méthode d'une autre classe définie ainsi    public void miseajour(){...}		Q 17
Quand t1 est en train d'exécuter cette méthode alors t2 (qui veut aussi l'exécuter) est en attente que t1 ait fini de l'exécuter		
1	OUI	
2	NON	

La classe Thread prédéfinie du langage Java est une classe abstraite		Q 18
1	OUI	
2	NON	

Le code suivant crée un fichier de nom "exemple.txt" dans le répertoire courant d'exécution. Le fichier est créé, vide de toute information		Q 19
<pre> File fichier; fichier = new File("exemple.txt"); </pre>		
1	OUI	
2	NON	

Soit le code suivant :		Q 20
<pre>public class Individu {String nom; String prenom;}  Individu ind = new Individu(); File fic = new File(".", "fic.bin"); FileOutputStream fin = new FileOutputStream(fic); ObjectOutputStream fich = new ObjectOutputStream(fin);  fich.writeObject((Object)ind);</pre>		
Ce code est correct		
1	OUI	
2	NON	

On a le code suivant :		Q 21
<pre>File fichier = new File("ListeDouble.bin"); FileOutputStream fos = new FileOutputStream(fichier); DataOutputStream dos = new DataOutputStream(fos); dos.writeInt(tab.length); for(int i=0;i&lt; tab.length;i++) dos.writeDouble( tab[i]); dos.close();</pre>		
1	Ce code crée un fichier de nom 'ListeDouble.bin' contenant la liste de doubles	
2	Ce code crée un fichier dont les informations sont dans un format texte	
3	Ce code crée un fichier dont les informations sont dans un format binaire	

La class File ne gère que les fichiers. Pour gérer des répertoires on utilise la classe Directory		Q 22
1	OUI	
2	NON	

La sérialisation est un service du langage Java qui permet d'écrire et de lire n'importe quel objet dans un fichier binaire (Serializable)		Q 23
1	OUI	
2	NON	

Le code suivant est correct :		Q 24
<pre>import java.io.*;  public class Terminal{     static BufferedReader in =         new BufferedReader(new InputStreamReader(System.in));      public static String lireString() throws IOException     {         return in.readLine();     } }</pre>		
1	OUI	
2	NON	

Le package est une unité de programmation permettant de regrouper et architecturer les classes du langage Java (prédéfinies ou développées) dans des répertoires et accessibles aux autres unités de programmation (programme Java, Applet, ...).		Q 25
1	OUI	
2	NON	

Un package est un répertoire qui contient les classes du package		Q 26
1	OUI	
2	NON	

Soit le fichier suivant C:\CodeJava\exercices\cnam\util\Terminal.java. Le fichier Terminal.java contient en 1 <sup>ère</sup> ligne : package exercices.cnam.util Le répertoire C:\bin est vide. On réalise la commande suivante : javac -d C:\bin C:\CodeJava\exercices\cnam\util\Terminal.java On obtient le résultat suivant : C:\bin\exercices\cnam\util\Terminal.class		Q 27
1	OUI	
2	NON	

En JAVA, un thread est une JVM qui s'exécute en parallèle de la JVM dans laquelle le thread a été créé.		Q 28
1	OUI	
2	NON	

Pour créer et démarrer un thread, il faut :		Q 29
1	créer un objet dont la classe d'appartenance hérite de la classe Runnable et implémente la méthode run	
2	créer un objet dont la classe d'appartenance hérite de la classe Thread, implémente la méthode run et appeler la méthode start	

La classe A hérite de B qui hérite de C. C est une classe abstraite qui implémente l'interface I. A et B ne sont pas des classes abstraites		Q 30
1	C peut implémenter une partie des méthodes de l'interface I	
2	B doit implémenter les méthodes de I qui n'ont pas été implémentées par A ou par C	
3	C ne peut pas implémenter des méthodes de l'interface I	

L'interface permet de :		Q 31
1	gérer des collections polymorphes (les éléments sont de type d'une interface)	
2	gérer les fichiers (interface avec le système d'exploitation)	
3	de passer en paramètre d'une méthode "un traitement" (traitements génériques)	

Un tableau Java peut contenir des objets dont la classe d'appartenance de ces objets est une classe abstraite		Q 32
1	OUI	
2	NON	

Soit le thread définit de la manière suivante : <pre>public class MonThread extends Thread { . . . } et MonThread t = new MonThread();</pre>		Q 33
On démarre le thread t de la manière suivante :		
1	t.run()	
2	t.start()	
3	new Thread(t).start()	

En JAVA, la classe RuntimeException qui hérite de la classe Exception permet de déclencher une exception qui interrompt proprement et définitivement le runtime de la JVM.		Q 34
1	OUI	
2	NON	

Le code suivant est correct :		Q 35
<pre>File f = new File("/etc/passwd"); System.out.println(f.exists()); System.out.println(f.canRead()); System.out.println(f.canWrite()); System.out.println(f.getLength());  File d = new File("/etc/"); System.out.println(d.isDirectory());  String[] files = d.list(); for(int i=0; i &lt; files.length; i++)     System.out.println(files[i]);</pre>		
1	OUI	
2	NON	

**(Tourner la page)**

## 2. Questions libres (15 points)

Chaque question est notée sur 5 points.

Vous répondez à ces questions sur une copie vierge en mettant bien le numéro de la question, sans oublier votre nom et prénom.

### Q 1

Nous avons les 4 classes suivantes : Livre Film Musique LivreNumerique qui définissent des types de média différents.

Expliquez ce qu'il est nécessaire de faire pour que un programme Java puisse gérer (afficher et trier) une collection de ces différents média.

Vous pouvez compléter vos explications par un diagramme de classe UML.

### Q 2

Expliquez ce qu'est la récursivité.

Donner un exemple (soyez précis).

### Q 3

Dans la programmation objet, expliquez à quoi sert la notion de classe abstraite.

Donner un exemple d'utilisation (soyez précis).

(Tourner la page)



---

**2<sup>ème</sup> PARTIE : PROGRAMMATION (avec document)**  
**Durée 1h15**

---

**Probleme 1 [15 points]**

En utilisant, notamment, les classes d'entrées sorties Java DataInputStream et DataOutputStream écrire le programme JAVA complet qui réalise séquentiellement les traitements suivants :

- charger en mémoire la configuration quelconque d'un jeu d'échec,
- changer la position d'une pièce (nouvelle saisie des coordonnées d'une pièce),
- enregistrer la nouvelle configuration du jeu d'échec.

Un jeu d'échec est un damier 8x8 dont les lignes sont numérotées de 1 à 8 et les colonnes de A à H.

Les pièces peuvent se trouver sur n'importe quelle case du damier (exemple : Tour noire en G5) et peuvent être soit un Roi, une Dame, un Fou, un Cavalier, une Tour ou un Pion. Chaque pièce peut être blanche ou noire.

**Probleme 2 [15 points]**

1/ Ecrire la classe ObservateurCyclique dont le rôle est d'observer cycliquement (thread) l'état d'un objet dont la classe d'appartenance implémente l'interface EtatCyclique.

Cette interface décrit trois méthodes :

- Etat getEtat() qui retourne l'état de l'objet
- int getTime() qui retourne le temps d'attente entre deux cycles
- boolean getStop() qui retourne true ou false permettant d'arrêter le thread d'observation.

Etat est une classe abstraite qui contient la méthode String toString() qui retourne l'état de l'objet surveillé.

Le thread d'observation affiche dans la console d'exécution l'état de l'objet surveillé.

2/ Soit la classe Cible qui implémente l'interface EtatCyclique.

Ecrire le programme principale qui observe l'état d'un objet de type Cible.

—

**(Fin du sujet)**