+75/1/23+

QCM

TEST

Introduction à la programmation					
orientée objet					
9/11/2017					

Nom et prénom :	
STROBBE Nathan	
Groupe: .2	

Cochez les cases en mettant une X.

Le symbole \bigoplus indique que la question peut avoir zéro, une ou plusieurs bonnes réponses. Pour ces questions, cocher une bonne réponse apporte des points positifs ; cocher une mauvaise réponse peut apporter des points négatifs. Dans tout le code, les package et les import sont censés être correctement déclarés. Toute classe est supposée être dans le bon package, dans le bon fichier, avec les bons import.

```
Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____?
Question 1
package toto;
public class Toto {
      _ String doSomething() {
       // does something
}
                                                      X private
     package-private (default)
                                                      public
                 Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____?
Question 2
                                                     package toto;
package toto;
public class Toto {
                                                     class Foobar {
       String doSomething() {
                                                        private Toto toto = new Toto();
       // does something
                                                        private String doSomething() {
                                                           toto.doSomething();
     private
                                                          public
 package-private (default)
                Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____?
Question 3
```

0.33/0.33

0.33/0.33

0.33/0.33

package-private (default)

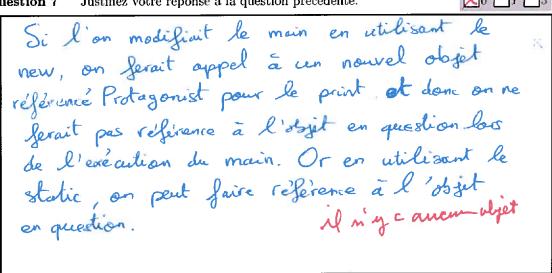
____ private

public



	Question 4 Quenes animations's appliquent aux maps (la classe Hashmap):							
1/1	sont initialsés par, eg, new Person(14) peuvent stocker des doublons (deux fois le même élément) sont dans le package java.util l'ordre de stockage des éléments est bien défini sont déclarés par, eg, Person[] p peuvent stocker des primitifs, eg, double le nombre d'éléments est donné par .size() sont de taille fixe sont indexés exclusivement par des entiers non-négatifs							
Question 5 Quelles affirmations s'appliquent aux listes (la classe ArrayLis								
1/1	 □ l'ordre de stockage des éléments est bien défini □ sont dans le package java.util □ sont de taille fixe □ peuvent stocker des primitifs, eg, double □ le nombre d'éléments est donné par .size() □ sont déclarés par, eg, Person[] p □ peuvent stocker des doublons (deux fois le même élément) □ sont créés par, eg, new Person(14) □ sont créés par, eg, new Person(14) □ sont créés par, eg, new Person(14) 							
	Question 6 Le code							
	<pre>class Protagonist { private String name = "Fred"; private void print() { System.out.println("My name is " + name); } public static void main(String args) { print(); }</pre>							
	ne compile pas. Le compilateur dit							
	Protagonist.java:11: error: non-static method print() cannot be referenced from a static context print();							
	1 error							
	Deux possibilités se présentent, laquelle est le meilleur choix :							
0/1	Modifier main : new Protagonist().print() Déclarer : private static void print()							

Question 7 Justifiez votre réponse à la question précédente.



```
Question 8 D Lesquelles des expressions déclarent, construisent et initialisent un tableau?
                                                               int[] myList = {"1", "2", "3"};
                int myList = \{4,9,7,0\};
2/2
                                                               int myList[] = {4, 3, 7};
                |\inf[] \text{ myList} = (5, 8, 2);
              Question 9 ( ) On souhaite écrire une application dans un package foobar. Il est obligatoire
              de...
                mettre tous les fichiers sources dans un même .jar
                compiler les fichiers avec la commande javac -package foobar *.java
0.5/0.5
                écrire import foobar.*; en début de tous les fichiers source
                Kécrire package foobar; en début de tous les fichiers source
              Question 10 
Nous souhaitons compiler le code source :
               package main;
                                                              package main;
                                                              class Toto ()
               class Main {
                  private Toto toto;
                  public static void main(String... args) {
                     // some code
```

Le code est dans le dossier myproject/src/main, et on aimerait que le bytecode soit généré dans le dossier myproject/bin. Tout en restant dans myproject, lesquelles des commandes feraient l'affaire :

```
javac -d bin src/main/*.java

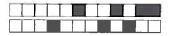
javac -d bin src/main/Main.java src/main/Toto.java

0.5/0.5

javac -d bin src/main/Main src/main/Toto

javac src/main/*.java

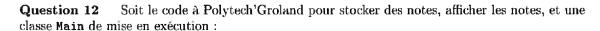
javac -d bin *.java
```



	Question 11 Pour exécuter le coo	de de la question précédente, ça serait :
	java -cp bin Main	java -cp bin Main.java
.5/0.5	🔀 java -cp bin main.Main	java -cp bin Main.class
	java main.Main	java bin/Main.class

or.

-40



```
package admin;
                                                                 package admin;
                                                                 class Consulter {
class Marks {
    // this is voodoo, but it correctly intializes marks
                                                                    private final Marks marks;
    private final Map<String, int[]> marks
       = new HashWap<String, int[]>(){{
put("Barney", new int[]{12, 8});
put("Fred", new int[]{7, 9});
put("Wilma", new int[]{15, 13});
                                                                    Consulter (Marks marks) {
                                                                        this.marks = marks:
                                                                    void displayMarks(String student) {
                                                                        System.out.print(student + ": ");
    int[] getMarks(String student) {
                                                                        for (int m : marks.getMarks(student)) {
                                                                            System.out.print(m + " ");
        return marks.get(student);
                                                                        System.out.println();
    Set<String> getStudents() {
       return marks.keySet();
                                                                package admin;
package admin.sploit;
                                                                public class Main {
                                                                    public static void main(String... args) {
public class Sploit {
                                                                        Marks marks = new Marks();
    // code to be supplied for the following method
                                                                        Consulter consulter = new Consulter(marks);
    public void haxMyMarks
                                                                        // administration consults student marks
                                                                        marks.getStudents().forEach(s
                                                                                -> consulter.displayMarks(s));
                                                                        // Wilma introduces exploit
                                                                        new Sploit().haxMyMarks(marks.getMarks("Wilma"));
                                                                        // administration consults student marks again
                                                                        marks.getStudents().forEach(s
                                                                                -> consulter.displayMarks(s));
                                                                    1
```

En fonctionnement normal, tout cela donne le résultat à gauche :

Barney: 12 8 Barney: 12 8 Wilma: 15 13 Wilma: 20 20 Fred: 7 9 Fred: 7 9

Hélas, une élève rusée a trouvé le moyen d'introduire du code dans la classe Sploit pour exploiter une faille dans le système, afin d'améliorer ses notes. Cela donne le résultat à droite. Démontrez comment elle aurait pu arriver à ce résultat en complétant la classe Sploit, sans toucher aux autres classes.

```
public void Rox My Harks (int[] marks) {

Sor(int mark: marks) {

mark = 20;

fauchail hurde aute

For(int i=0;...;...)

marks[i = 20;
```

3/3

Question 13 Soit le code de la question précédente. Quelle parade dans la classe Marks, et seulement dans la classe Marks, aurait pu éviter ce désagrément pour Polytech'Groland?

Il Jaudrait que le tableau d'entiers

dans la map soit "final". Cela éviterait

de pouvoir changer ces notes à partir d'outres desses

Donc par exemple e put ("Barney", new final int[]

[12]

son contem pouvoir

ty ître modifie

Question 14 Soit la déclaration : private final String[] names = {"Fred"}; Quelles expressions sont permises dans une méthode de la même classe? names[0] == "Barney"; names = new String[1]; 0/2 \mathbf{X} names[0] = "Barney"; names = "Barney"; names = new String("Barney"); Question 15 (Soit la déclaration : private String[] names = {"Fred"}; Quelles expressions sont permises dans une méthode de la même classe? __ names = "Barney"; names = new String("Barney"); names[0] = "Barney"; 1.33/2 \mathbf{X} names = new String[1]; names[0] == "Barney";

Question 16 Corrigez toutes les erreurs dans le code :

```
Void print Greater (double [] marks, double mean) {

for (int index=0; index < marks-length; index ++) {

if (marks [index] > mean) {

System.out.println(marks [index]);

}

}
```

Question 17 \bigoplus Quelles affirmations s'appliquent aux tableaux :

	le nombre	d'éléments	est	donné	par	
	.size()					
X	peuvent stocker des doublons (deux fois le même élément)					
	sont créés pa	ar, eg, p = n	ew P	erson(1	14)	
	sont dans le	package jav	a.ut	il		
X	penvent stoo	ker des prim	itifs.	eg, dou	ble	

1/1

- X sont de taille fixe
- 🔀 sont déclarés par, eg, Person[] p
- l'ordre de stockage des éléments est bien défini
- sont indexés exclusivement par des entiers non-négatifs

1/1