***Université Mohamed Boudiaf - M’sila***

***Faculté  Technologie***

***Département électronique***

***Examen POO JAVA Ch. Cours :B.Mourad M2. ESEM Durée 1h30***

Nom : ………………………………….. Prénom : ……………..………………… Note ……………..

***Questions de cours (04pts)* :**

***A) compléter les définitions suivantes :***

Une instance de classe est un ………………………………………………………………………………………………

Quel est le but de cette commande : **java –noasyncgc** ……………………………………………………………….

***b) Compléter le tableau suivant en Citant deux différences majeurs entre C++ et JAVA***

|  |  |
| --- | --- |
| ***C++*** | ***JAVA*** |
|  |  |
|  |  |

***Exercice 2 (06pts –P.O.O)***

Créer une classe permettant de gérer les nombres complexes. Cette classe représentant un nombre complexe doit permettre l’affichage sous forme ( x + i y) ainsi que deux méthodes une pour l'addition et l’autre pour la soustraction de ce nombre avec un autre nombre complexe (chaque méthode retourne un nouveau objet de cette classe) ? Tester cette classe avec des valeurs de votre choix ?

***Exercice 3(05pts)***: *Cocher La bonne Réponse ?*

|  |  |
| --- | --- |
| Java est un langage développé par Oracle  | OUI NON  |
| Le bytecode est un code exécuté par la machine sans aucun intermédiaire  | OUI NON  |
| un attribut ***protected*** est accessible uniquement aux classes d'un package.  | OUI NON  |
| une méthode de classe peut être appelée sans même qu'il n'existe d'instance | OUI NON  |
| une méthode de classe peut accéder à des attributs non statiques. | OUI NON  |

***Exercice 4 (05 pts)***: *programmation java*

Ecrire un programme Java qui calcule le factoriel d’un nombre entier positif saisi au clavier. ( N ! = n \* (n-1) \* . . . \* 1)

 *Ecrire le programme au verso* ***Bonne chance***

***Université Mohamed Boudiaf - M’sila***

***Faculté  Technologie***

***Département électronique***

***Examen POO JAVA Ch. Cours :B.Mourad M2. ESEM Durée 1h30 2019/2020***

Nom : ……………Corr…………………….. Prénom : type Note ……………..

***Questions de cours (06pts)* :**

***A) compléter les définitions suivantes :***

Une instance de classe est un ………………. objet concret

Quel est le but de cette commande : **java –noasyncgc** …………… désactiver le garbage collector (gc)

***b) Compléter le tableau suivant en Citant deux différences majeurs entre C++ et JAVA***

|  |  |
| --- | --- |
| ***C++*** | ***JAVA*** |
| Permet la programmation procédurale et P.O.O | P.O.O pure |
| La compilation Produit des fichiers exécutables directement par les machines  | La compilation produit un code intermédiaire (\*.class) qui est exécutable par le JVM |

***Exercice 2 (04pts –P.O.O)***

**class** Complex **{**

 **float** reel; // Partie réelle

 **float** img; // Partie imaginaire

 Complex(**float** r, **float** im) **{** reel = r; img = im;**}** // Constructeur

 **void** afficher(Complex c) **{** System.out.println(c.reel+"i"+c.img);

 Complex plus(Complex c1, Complex c2) **{**

 **return** **new** Complex(c1.reel+c2.reel , c1.img+c2.img);}

 Complex minus(Complex c1, Complex c2) **{**

 **return** **new** Complex(c1.reel-c2.reel , c1.img-c2.img);}

 public static void **main**(String[] args) {

 **Complex** a = new **Complex**(5.0, 6.0); afficher(a);

 **Complex** b = new **Complex**(-3.0, 4.0); afficher(b);

 **Complex** c = a.plus(b); afficher(c);

 **Complex** d = a.minus(c); afficher(d); }

**}**

***Exercice 3(05pts)***: *Cocher La bonne Réponse ?*

|  |  |
| --- | --- |
| Java est un langage développé par Oracle  | OUI **NON**  |
| Le bytecode est un code exécuté par la machine sans aucun intermédiaire  | OUI **NON**  |
| un attribut ***protected*** est accessible uniquement aux classes d'un package.  | OUI **NON**  |
| une méthode de classe peut être appelée sans même qu'il n'existe d'instance | **OUI** NON  |
| une méthode de classe peut accéder à des attributs non statiques. | OUI **NON**  |

***Exercice 4 (05 pts)***:

**import** java.util.Scanner;

**public** class Fact {

 **public** static void main (String[] args) {

 int f = 1;

 int n;

 **Scanner** sc = new **Scanner**(System.in);

 **System**.out.print("entrez un entire! : ");

 n = sc.nextInt();

 for (int i=1; i<=n; i++)

 f=f\*i;

 System.out.print(f);

 }