Interrogation AP1 – 10 octobre 2013

Lisez attentivement l'ensemble de l'énoncé avant de répondre et veillez à écrire proprement Aucun document et aucune machine ne sont autorisés

Exercice 1: vive les conditions

Complétez le tableau en inscrivant T pour true ou F pour false dans chacune des cases :

	a = 3; b = 'y'	a = 5; b = 'b'	a = 7; b = 'h'
(a < 5)			
(b < 'f')			
(a < 5) && (b < 'f')			
!(a < 5) (b < 'f')			
(a < b) ^ (b < 'f')			

Exercice 2: patchwork

Les questions qui suivent sont indépendantes, vous pouvez donc répondre seulement à certaines d'entre elles si vous êtes bloqués sur une des questions.

On suppose que l'on dispose des fonctions de comparaison selon l'ordre lexicographique :

```
// test si les chaînes s1 et s2 sont égales
boolean equals(String s1, String s2)
// test si s1 est supérieure ou égale à s2
boolean greaterThanOrEqual(String s1, String s2)
// test si s1 est inférieure ou égale à s2
boolean lessThanOrEqual(String s1, String s2)
```

Q1 : On souhaite disposer d'une fonction permettant de déterminer quelle est la plus petit chaîne de caractères parmi deux chaînes. Voici les tests vérifiant la validité de cette fonction :

```
void testMin() {
  assertEquals("Godel" , min("Turing", "Godel") ;
  assertEquals("Godel" , min("Godel", "Godot") ;
}
```

Donnez la signature et le corps de la fonction min :

On suppose dans la suite que l'on dispose de la fonction symétrique max qui retourne la chaîne la plus grande.

Q2 : A l'aide des fonctions min et max, donnez les expressions permettant de déterminer la plus petite chaîne, respectivement la plus grande chaîne, parmi trois variables a, b et c :

```
void algorithm() {
  String a = "toto", b = "titi", c = "tata";
  String plus_petite =
  String plus_grande =
}
```

ATTENTION : vos expressions doivent bien évidemment fonctionner quelque soient les valeurs contenues par les variables a, b et c !

Q3 : On dispose d'un ensemble de lettres repérables par leurs coordonnées de ligne et de colonne :

	c = 0	c = 1	c = 2	c = 3	c = 4
l = 0	'a'	'a'	'b'	'b'	'a'
l = 1	'c'	'c'	'b'	'b'	'c'
l=2	'c'	'c'	'b'	'b'	'c'
<i>l</i> = <i>3</i>	'a'	'a'	'a'	'a'	'a'

Donnez le corps de la fonction lettre qui retourne la lettre correspondant aux coordonnées de ligne et colonne passés en paramètre : char lettre (int ligne, int colonne) **ATTENTION** : Vous ne pouvez utiliser que des alternatives sans partie sinon (ie. pas de else!) et votre fonction ne doit utiliser qu'une seule fois l'instruction return.

Q4: On souhaite maintenant afficher une partie de la table ASCII. Donnez l'algorithme (void algorithme) permettant de produire l'affichage suivant :

```
65 = A, 66 = B, 67 = C, 68 = D, 69 = E, 70 = F, 71 = G, 72 = H, 73 = I, 74 = J, 75 = K, 76 = L, 77 = M, 78 = N, 79 = O, 80 = P, 81 = Q, 82 = R, 83 = S, 84 = T, 85 = U, 86 = V, 87 = W, 88 = X, 89 = Y, 90 = Z
```

ATTENTION: Seules les questions 3 à 4 du deuxième exercice doivent être rendues sur la copie. Pour les autres, répondez directement sur l'énoncé.

Il y a un autre exercice au verso!

Exercice 3: un dédale d'alternatives

Un développeur peu scrupuleux a écrit un algorithme peu lisible et assez complexe. Dans un premier temps, dessinez l'organigramme de cet algorithme (sur la page suivante) pour mieux le comprendre.

```
class Dedale extends Program{
 String f(int a, int b, int c) {
   String rep = "" ;
  if (a > c) {
2
     if (b == c) {
3
        rep = rep + "bon";
     } else if (a >= c) {
5
        rep = rep + "ta";
6
     }
7 } else {
    if (a < b) {
8
        rep = rep + "ta";
     } else {
10
        rep = rep + "tu";
11
12
13
     rep = rep + "pe";
14 }
15 return rep;
 }
}
```

Pour chaque entrée (donnée par a, b et c), indiquez dans le tableau ci-dessous le résultat de la fonction f. **ATTENTION** : il peut y avoir plusieurs cases cochées sur une même ligne ...

f(a,b,c)	0,3,5	5,4,3	1,2,1	3,4,4	6,5,5	4,4,4	Impossible avec les valeurs précédentes	Impossible quelque soit les valeurs
"bon"	0	0	0	0	0	0	0	0
"bonta"	0	0	0	0	0	0	0	0
"ta"	0	0	0	0	0	0	0	0
"tu"	0	0	0	0	0	0	0	0
"tatu"	0	0	0	0	0	0	0	0
"tape"	0	0	0	0	0	0	0	0
"tupe"	0	0	0	0	0	0	0	0
11 11	0	0	0	0	0	0	0	0

NOM:	PRENOM:	GROUPE
	<organigramme du="" dédale="" programme=""></organigramme>	

ATTENTION

Mettez l'énoncé de l'interrogation dans votre copie et de vérifiez bien que vous avez inscrit vos nom, prénom et groupe avant de rendre votre copie!