

Session du 31 janvier 2013

Aucun document n'est autorisé. Le barème est donné à titre indicatif.

Exercice 1 : exécution de programme (3 points)

Retracez l'exécution du programme suivant en indiquant l'évolution des valeurs des variables manipulées par ce programme. Vous pourrez faire cela au moyen d'un tableau comportant une colonne pour chaque variable, en précisant sur chaque ligne du tableau le numéro de ligne de code exécutée.

```
1 public class Ex13_3{
2     public static void main(String [] args){
3         int x = 0;
4         String st = "abc";
5         boolean bool = true;
6         char ch1 = 'b';
7         char ch2 = 'x';
8         for (int i=0; i<5; i++){
9             bool = (i % 2 == 0);
10            if (bool == (i<3)){
11                st = st + ch1;
12            }else{
13                st = st + ch2;
14            }
15        }
16        Terminal.ecrireStringln("valeur de st : " + st);
17    }
18 }
```

On rappelle que % est l'opérateur modulo qui calcule le reste de la division entière. L'expression $(i \% 2 == 0)$ vaut true si i est pair et false si i est impair. Terminal.ecrireStringln affiche une chaîne de caractères à l'écran.

1 point si la réponse est adéquate dans sa forme et les valeurs initiales des variables sont correctes, 1 point si les instructions de la boucles sont exécutées 5 fois, 1 point si les valeurs finales des variables sont bonnes. La présentation suivante comprend des éléments non demandés dans l'énoncé (valeur des conditions et écran) et ne doivent donc pas intervenir dans la notation.

lig	x	st	bool	ch1	ch2	i	i;5	bool == (i;3)	écran
3	0	X	X	X	X	X			
4	0	`` abc ``	X	X	X	X			
5	0	`` abc ``	true	X	X	X			
6	0	`` abc ``	true	'b'	X	X			
7	0	`` abc ``	true	'b'	'x'	X			
8.init	0	`` abc ``	true	'b'	'x'	0			
8.test	0	`` abc ``	true	'b'	'x'	0	true		
9	0	`` abc ``	true	'b'	'x'	0			
10	0	`` abc ``	true	'b'	'x'	0		true	
11	0	`` abcb ``	true	'b'	'x'	0			
8.incr	0	`` abcb ``	true	'b'	'x'	1			
8.test	0	`` abcb ``	true	'b'	'x'	1	true		
9	0	`` abcb ``	false	'b'	'x'	1	true		
			...						
8.test	0	`` abcbxbbx ``	false	'b'	'x'	5	false		
16	0	`` abcbxbbx ``	false	'b'	'x'	5	false		abcbxbbx

Exercice 2 : tableau de caractères (4 points)

Ecrivez un programme (méthode `main`) qui réalise successivement les opérations suivantes :

1. **1 point** Déclaration et initialisation d'un tableau de 10 caractères en lisant les 10 valeurs au clavier.
2. **1,5 point** Vérification que toutes les valeurs du tableau sont des lettres. Pour cela on pourra utiliser la méthode `Character.isLetter` qui prend en paramètre un caractère et renvoie `true` si et seulement si ce caractère est une lettre.
3. **1,5 point** Calcul du nombre de majuscules et de minuscules dans le tableau. Vous pourrez utiliser la méthode `Character.isLowercase` qui prend en paramètre un caractère et renvoie `true` si et seulement si ce caractère est une lettre en minuscule, et la méthode `Character.isUppercase`

```

public class Ex13_6{
    public static void main(String [] args){
        char [] tab = new char [10];
        for (int i=0; i<10; i++){
            Terminal.ecrireStringln ("Entrez un caractère");
            tab[i] = Terminal.lireChar ();
        }
        boolean touslet = true;
        for (int i=0; i<10; i++){
            if (!Character.isLetter (tab [i])){
                touslet = false;
            }
        }
        if (touslet){
            Terminal.ecrireStringln ("Tous les caractères sont des lettres");
        } else {
            Terminal.ecrireStringln ("Tous les caractères ne sont pas des lettres");
        }
    }
}

```

```

    int nbmaj = 0, nbmin = 0;
    for (int i=0; i<10; i++){
        if (Character.isUpperCase(tab[i]))
            nbmaj++;
        else if (Character.isLowerCase(tab[i]))
            nbmin++;
    }
    Terminal.ecrireStringln("nombre_de_majuscules:" +nbmaj);
    Terminal.ecrireStringln("nombre_de_minuscules:" +nbmin);
}
}

```

Exercice 3 : procédures et fonctions (3 points)

Donnez les affichages produits à l'exécution par le programme suivant :

```

public class ExoMethodes3{
    static int toDec(boolean b) {
        if (b) return 1; else return 0;
    }
    static void ch(int i, boolean b, boolean [] t) {
        if (0<= i && i <= t.length-1)
            t[i] = b;
    }
    static int eval(boolean [] tb) {
        int s = 0;
        int p = 1;
        for (int i=tb.length-1; i>=0; i--) {
            s = s + toDec(tb[i])*p;
            p= p*2;
        }
        return s;
    }
    static void af (boolean [] t) {
        for (int i=0; i<t.length; i++) {
            Terminal.ecrireString("_" + t[i]);
        }
        Terminal.sautDeLigne();
    }

    public static void main (String args[]) {
        boolean [] tab = {false, false, true};
        ch(3,true, tab);
        af(tab);
        ch(0,true, tab);
        af(tab);
        Terminal.ecrireStringln("resultat_=_"+ eval(tab));
    }
}

```

```

false false true
true false true
resultat = 5

```

Exercice 4 : élections (5 points)

On souhaite utiliser un programme pour enregistrer les résultats d'une élection. Ces résultats donneront le nombre de voix obtenu par candidat et par bureau de vote. Les candidats sont Riri, Fifi et Loulou. Il y a 5 bureaux de votes.

1. **1,5 point** définissez la ou les variables nécessaires à contenir ces données ainsi que le code nécessaire pour lire au clavier les données sur le vote.
2. **1,5 point** écrivez une méthode qui calcule le pourcentage d'un candidat donné. Cette méthode ne doit rien afficher.
3. **1 point** utilisez cette méthode pour faire afficher le pourcentage de vote obtenus par chacun des candidats.
4. **1 point** affichez le nom du vainqueur.

Accepter sans pénalité les pourcentages de type double ou int et le codage des noms de candidats par des entiers (par exemples dans les arguments de la méthode).

```

public class Ex13_7{
    public static void main(String[] args){
        int[][] reselec = new int[3][5];
        for (int i=1; i<6; i++){
            Terminal.ecrireStringln("Entrez le nombre de voix pour fifi"
                + " dans le bureau numero" + i);
            reselec[0][i-1]=Terminal.lireInt();
        }
        for (int i=1; i<6; i++){
            Terminal.ecrireStringln("Entrez le nombre de voix pour riri"
                + " dans le bureau numero" + i);
            reselec[1][i-1]=Terminal.lireInt();
        }
        for (int i=1; i<6; i++){
            Terminal.ecrireStringln("Entrez le nombre de voix pour loulou"
                + " dans le bureau numero" + i);
            reselec[2][i-1]=Terminal.lireInt();
        }
        int pcf, pcr, pcl;
        pcf= pourcent(reselec,"fifi");
        pcr = pourcent(reselec,"riri");
        pcl = pourcent(reselec,"loulou");
        Terminal.ecrireStringln("Pourcentage de fifi"+ pcf);
        Terminal.ecrireStringln("Pourcentage de riri"+ pcr);
        Terminal.ecrireStringln("Pourcentage de loulou"+ pcl);
        if (pcf > pcr && pcf > pcl){
            Terminal.ecrireStringln("Vainqueur fifi");
        }else if (pcr > pcl){

```

```

        Terminal. ecrireStringln("Vainqueur_riri");
    } else {
        Terminal. ecrireStringln("Vainqueur_loulou");
    }
}
public static int pourcent(int[][] tab, String nom){
    int indnom, nbexprimes = 0, nbcandidat = 0;
    if (nom.equals("fifi")){
        indnom=0;
    } else if (nom.equals("riri")){
        indnom=1;
    } else {
        indnom=2;
    }
    for (int i = 0; i<3; i++){
        for (int j=0; j<5; j++){
            nbexprimes = nbexprimes+tab[i][j];
            if (i==indnom)
                nbcandidat = nbcandidat + tab[i][j];
        }
    }
    return (nbcandidat*100)/nbexprimes;
}
}

```

Exercice 5 : nombres à virgule (5 points)

Un nombre à virgule est tapé au clavier au moyen d'une suite de caractères, un par ligne. Il peut y avoir des espaces avant le début et après la fin du nombre. Les chiffres sont entrés sur des lignes consécutives. La partie décimale est séparée par une virgule (notation française). Tout caractère autre qu'un chiffre, un espace ou une virgule est une erreur.

Voici un exemple d'entrée correcte du programme :

```

5
6
,
7
0
5

```

Ecrivez un programme (méthode main) qui transforme une telle entrée en une valeur de type double stockée dans une variable et affichée à la fin du programme.

Chaque caractère peut être lu par la méthode `Terminal.lireChar` ou en utilisant un scanner.

Un caractère représentant un chiffre peut être converti en un nombre entier au moyen de la méthode suivante `Character.getNumericValue` qui prend en argument un caractère et renvoie l'entier correspondant.

On ne demande pas de traiter les erreurs, notamment si les caractères lus ne forment pas un nombre à virgule.

```
public class Ex13_8{
    public static void main(String[] args){
        int partieEntiere = 0;
        double partieDec = 0.0;
        double diviseur = 10.0;
        double res;
        char c;
        do{
            c = Terminal.lireChar();
        }while(c!='\n');
        while (c != ','){
            partieEntiere = partieEntiere * 10 + Character.getNumericValue(c);
            c = Terminal.lireChar();
        }
        c = Terminal.lireChar();
        while(c!='\n'){
            partieDec = partieDec + (Character.getNumericValue(c)/diviseur);
            c = Terminal.lireChar();
            diviseur = diviseur * 10;
        }
        res = partieEntiere + partieDec;
        Terminal.ecrireDoubleLn(res);
    }
}
```

1,5 point pour le calcul de la partie entiere, 1,5 point pour le calcul de la partie décimale, 1 point pour la bonne gestion des délimiteurs, 1 point pour le reste