

**Partiel de l'UE INF f1 – Novembre 2016 - 1h40**  
**Une feuille A4 autorisée**

*Il est inutile d'indiquer les premières lignes des programmes Java :  
import..., ...class..., ...main... En revanche, les entêtes des actions et fonctions sont  
requis.*

**Exercice 1 (4 pts, environ 20 minutes)**

a) Écrire 1001,01 en base 10 et en base 16.

$$= 9,25 = (9,4)_{16}$$

b) Écrire -8 sur un octet en utilisant la méthode du complément vrai.

$$8=(00001000)_2 \text{ donc } -8 \text{ est représenté par } 11111000$$

c) On suppose que les réels sont représentés sur 3 octets avec un bit de signe et 7 bits d'exposant. Comment est représenté le nombre  $(EB1,6D9)_{16}$  ? Quel nombre a-t-on effectivement représenté ?

En écrivant chaque chiffre en base 16 sous la forme de 4 chiffres binaires, on obtient  
111010110001,011011011001 =  $1,11010110001,011011011001 \times 2^{11}$

Le signe est positif donc le bit de signe est à 0

L'exposant est 11, auquel il faut ajouter l'excédent qui vaut  $2^{7-1}-1 = 63$ , ce qui donne 74 qui s'écrit 1001010 sur 7 bits

La mantisse ne peut contenir que les 16 premiers bits après la virgule.

Cela donne donc : 010010101101011000101101

On a perdu les 7 derniers chiffres donc on n'a représenté que 111010110001,01101 qui est EB1,68.

**Exercice 2 (5 points, environ 25 minutes)**

Écrire une fonction qui, étant donnés une chaîne de caractères et deux indices, renvoie un booléen indiquant si les caractères à ces deux indices sont les mêmes. Par exemple, avec la chaîne "MIASHS" et les deux entiers 3 et 5, la fonction renverra true.

```
static boolean memesIndices(String ch, int i1, int i2) {  
    return(ch.charAt(i1)==ch.charAt(i2));  
}
```

**Exercice 3 (5 points, environ 25 minutes)**

Écrire un programme

a) qui remplit un tableau de 1000 flottants avec des valeurs flottantes aléatoires entre 0 et 1 (exclus) ;

b) et qui calcule et affiche la moyenne de ces valeurs.

```
float t[]=new float[1000];
int i=0;
float somme =0;
while (i<1000) {
    t[i]=(float)Math.random();
    somme += t[i];
    i++;
}
System.out.println("Moyenne : "+(float)somme/1000);
```

#### **Exercice 4 (6 points, environ 30 minutes)**

Écrire une action qui, étant donné une chaîne de caractères de longueur quelconque, affiche toutes les paires de caractères. Par exemple, avec la chaîne "PAF", le programme devra afficher :

PP - PA - PF - AP - AA - AF - FP - FA - FF -

Avec "ABCD", le programme devra afficher :

AA - AB - AC - AD - BA - BB - BC - BD - CA - CB - CC - CD - DA - DB - DC - DD -

```
public static void affPaire(String ch) {
    int i,j;
    i=0;
    while(i<ch.length()) {
        j=0;
        while (j<ch.length()) {
            System.out.print(""+ch.charAt(i)+ch.charAt(j)+"-");
            j++;
        }
        i++;
    }
}
```