

# INF1120 Programmation I

Hiver 2009

## Examen Intra

14 mars 2009

### CONSIGNES

Les règlements de l'UQAM concernant le plagiat seront strictement appliqués.

- . Il est important de bien expliquer vos choix s'il y a lieu.
- . Les surveillants ne répondront à aucune question.
- . Aucun document n'est permis sauf une feuille 8,5 " x 11 ".
- . La durée de l'examen est de 3 heures, de 14h00 à 17h00.
- . Vous pouvez utiliser les versos comme brouillons ou comme espace supplémentaire.
- . Les téléphones cellulaires, calculatrices, ordinateurs, palm, baladeurs, iPods, etc. sont interdits.

**Il est interdit de dégrafer le questionnaire.**

**Aucune sortie n'est permise durant l'examen.**

**Vous devez présenter votre carte d'étudiant sur demande**

### IDENTIFICATION

PRÉNOM: .....

NOM: .....

CODE PERMANENT: \_ \_ \_ \_ \_

SIGNATURE: .....

GROUPE: .....

PROFESSEUR: .....

#1 \_\_\_\_\_ / 8

#2 \_\_\_\_\_ / 12

#3 \_\_\_\_\_ / 16

#4 \_\_\_\_\_ / 12

#5 \_\_\_\_\_ / 10

#6 \_\_\_\_\_ / 10

#7 \_\_\_\_\_ / 15

#8 \_\_\_\_\_ / 17

TOTAL

\_\_\_\_\_ / 100

COMMENTAIRES

## Question 1 (8 points)

Répondez par Vrai ou Faux à chacune des affirmations suivantes (entourez V ou F).

- V    F    a)    Il est permis d'insérer une instruction `switch` dans une instruction `if`.
- V    F    b)    Un programme qui plante à l'exécution est un programme qui ne compile pas.
- V    F    c)    Toute instruction `if` doit comporter la partie `else`.
- V    F    d)    L'instruction `System.out.print(2 + 'A')` affichera la chaîne `2A`.
- V    F    e)    Le compilateur Java peut compiler une classe qui ne contient aucune variable.
- V    F    f)    Une constante déclarée dans une méthode ne peut être utilisée à l'extérieur de celle-ci.
- V    F    h)    Les deux boucles suivantes sont équivalentes (ont le même effet sur la valeur de la variable x):  
`for ( int i = 5; i < 10; i++ ) {x = x + 1;}`  
`for ( int i = 10; i > 5; i++ ) {x = x + 1;}`
- V    F    i)    Les instructions à l'intérieur d'une boucle `for` s'exécutent toujours au moins une fois.

## Question 2 (10 points : 1 point réponse)

Considérez les déclarations suivantes :

```
int      i = 6;  
double   d = 6.0;  
char     c = 'A';  
long     q = 3;  
float    t = 1.5f;  
boolean  b = false;
```

Indiquez le type et la valeur de chacune des expressions suivantes :

	Type	Valeur
a) <code>3.0 + 4 * 2</code>	_____	_____
b) <code>q * 3 + i / 2</code>	_____	_____
c) <code>t + d / i - q</code>	_____	_____
d) <code>(q &lt; 3)    ((b == true) &amp;&amp; (c != 'a'))</code>	_____	_____
e) <code>(byte)q + i * (short)(t - 1.5) + (byte)t</code>	_____	_____

### Question 3 (15 points : 3 points par sous-question)

À la suite de l'exécution de chacune des boucles suivantes, indiquez la valeur affichée (les 5 blocs sont indépendants les uns des autres). S'il s'agit d'une boucle infinie, indiquez-le.

<p>a)</p> <pre>int n = -2;  for ( int r = 7; r &gt; 3 ; r-- ) {     n = n + ( 7 - r ); }  System.out.println( n );</pre>	<p>b)</p> <pre>int x = 3; int p = 9;  for ( int i = 9; i &gt;= x; i-- ) {      do {         p = p - 1;         x = x + 1;      } while ( p != x &amp;&amp; i &gt;= x - 1 );      p = p + i; }  System.out.println( p );</pre>
<p>c)</p> <pre>int s = -11; int k = 13;  while ( k &gt;= -s ) {     k = k - 2;     s = s + 3; }  System.out.println ( k - s );</pre>	<p>d)</p> <pre>int x = 9;  do {     x = x + 3; } while ( x &gt;= 12 );  System.out.println(x);</pre>
<p>e)</p> <pre>int x = 5; int y = 0; do {     while ( x &lt; 10 ) {         x = x + 1;     }     y = y + 5; } while ( y &lt; 19 ); System.out.println ( x + y );</pre>	<p>Réponses :</p> <p>a) _____</p> <p>b) _____</p> <p>c) _____</p> <p>d) _____</p> <p>e) _____</p>

#### Question 4 (12 points : 3,5; 2,5; 6)

Considérer l'algorithme suivant :

```
debut
  Afficher "Entrez un nombre entier : "
  Saisir n (un entier)
  si n est supérieur ou égal à 10000 ou négatif alors
    Afficher "Nombre non valide"
  sinon
    si n est strictement supérieur à 999 alors
      machin <- 2
    sinon
      machin <- 1
    fin si
    si n est strictement supérieur à 99 alors
      machin <- machin + 1
    fin si
    si n est strictement supérieur à 9 alors
      machin <- machin + 1
    fin si
    afficher machin
  fin si
fin
```

(a) Donnez ce qui sera affiché pour chacune des valeurs de n saisies suivantes :

Valeur saisie n	Affiché à la console
-5	
0	
7	
654	
78	
8645	
1000	

(b) Dites en une phrase ce que l'algorithme fait en général (ne pas expliquer le code).

..... .....
----------------

(c) Traduire en Java le pseudo-code entre debut et fin

**(Utilisez la page suivante)**



### Question 5 (10 points)

Vous devez écrire une méthode qui **affiche** le numéro du jour de la semaine déterminé à partir d'un jour de référence et d'un décalage dans le temps. Les jours de la semaine sont numérotés à partir de 1 et le numéro de dimanche est 1. Exemples d'utilisation :

```
q5(1,4) affiche : Le numéro du jour de la semaine est : 5
           indice : 4 jours après le premier jour de la semaine, donc le jour numéro 5
```

```
q5(7,9) affiche: Le numéro du jour de la semaine est : 2
           indice : 9 jours après le septième jour de la semaine, donc le jour numéro 2
```

```
q5(3,0) affiche: Le numéro du jour de la semaine est : 3
           indice : 0 jours après le troisième jour de la semaine, donc le jour numéro 3
```

```
q5(3,-5) affiche : Le numéro du jour de la semaine est : 5
           indice : 5 jours avant le troisième jour de la semaine, donc le jour numéro 5
```

```
q5(10,2) affiche : Le numéro du jour de référence n'est pas valide
```

```

/**
 * paramètre numeroJourDeLaSemaine le numéro du jour de la semaine (1 = dimanche, 7 = samedi)
 * paramètre decalage               le nombre de jours de décalage à partir du jour de la semaine.
 * Affiche le numéro correspondant au jour de la semaine en fonction de numeroJourDeLaSemaine et
 * de decalage.
 */

```

```
public static void ..... à compléter
```



## Question 6 (10 points)

Trouvez les 5 erreurs de compilation qui se cachent dans le code source Java suivant. Pour chacune, indiquez le numéro de ligne où se situe l'erreur et la correction à apporter.

```
1.    public class ErreurCompilation {
2.        public static void main (String[] args) {
3.            short court = 3;
4.            double i = 4.0;
5.            FINAL float d = 1.0F;
6.            byte octet = 9.0;
7.            do {
8.                for (int k = 9; k < k; k--)
9.                    i = d + (long)k;
10.                   System.out.println(k + 'b');
11.                }
12.                octet = octet + "2";
13.            while (court < octet)
14.        } /** main */
15.    } /* ErreurConpilation */
```

Réponses :

	No ligne	Correction
1)	_____	_____ _____
2)	_____	_____ _____
3)	_____	_____ _____
4)	_____	_____ _____
5)	_____	_____ _____



## Question 7 (15 points)

À la suite de l'exécution des sélections suivantes, indiquez la valeur affichée par chacune des instructions  
System.out.println.

```
public static void main (String[] params) {
    int x = 5;
    int i = 10;
    int k = 3;
    long a = 10;
    double d = 9.2;
    boolean valide = false;
    float m = 7.1f;
    String message = "rien";
    if ( x + d <= 14 ) {
        if ( a % x == 0 ) {
            a = 0;
        }
    }
    else
        a = a * 2;
    a = a + 2;
    System.out.println ( a ); // reponse : _____

    if ( !( x != 3 && !valide ) != true ) {
        message = "ok";
    } else {
        message = "ko";
    }
    System.out.println ( message ); // reponse : _____

    switch ((int) m) {
        case 3 : m = x % i;
        case 1 : m = x + 1;
        case 2 : m = k + 5;
        case 4 : i = i / 2;
        case 5 : m = m + 9;
        case 6 : m = x + i;
        case 8 : m = m + 1;
        case 7 : m = m + (int) (m + 10) / 10;
        default : m = m + 1;
    }
    System.out.println(m); // reponse : _____

    if ( x == 6 ) {
    } else {
        switch (x + 2) {
            case 7 : break;
            case 6 : x = x + 2;
            case 5 : x = x + 1;
        }
    }

    System.out.println ( x ); // reponse : _____

    if ( d - m != -m + d ) {
        d = 100; }
        m = 17.1f;
    System.out.println ( (int) ( d + m ) ); // reponse : _____

} // main
```

## Question 8 (20 points)

Écrire un petit programme Java (méthode `main`) qui lit 3 nombres entiers: `depart`, `pas` et `limite`. Les valeurs de `pas` et de `limite` doivent être validées :

- La valeur de `pas` doit être non nulle.
- Si `pas` est positif, la valeur de `limite` doit être strictement supérieure à `depart`.
- Si `pas` est négatif, la valeur de `limite` doit être strictement inférieure à `depart`.

Le programme doit faire certains calculs à partir de ces valeurs. Voici les étapes :

### Étape 1- saisies et validations :

Votre programme devra **saisir** la valeur de `depart` (qui n'a pas à être validée). Ensuite, le programme saisit et valide la valeur de `pas`. Lorsque `pas` est valide, le programme doit saisir et valider la valeur de `limite`.

### Étape 2- calculs :

Une fois que les valeurs sont validées, le programme devra ajouter à `depart` la valeur de `pas` le plus de fois possible. La somme cumulée ne doit pas être supérieure à `limite` lorsque `pas` est positif ou être inférieure à `limite` dans le cas où `pas` est négatif.

### Étape 3 - affichage des résultats :

Enfin, le programme devra afficher la valeur de la somme ainsi que le nombre de pas ajoutés.

Exemples :

depart	pas	limite	somme	nombre de pas	détail du calcul
10	3	23	22	4	10 + 3 + 3 + 3 + 3
45	-4	19	21	6	45 + -4 + -4 + -4 + -4 + -4 + -4
100	26	110	100	0	100
-16	2	99	98	57	16 + 2 + 2 + 2 + ... + 2

Prenez pour acquis que l'utilisateur entrera uniquement des valeurs entières (pas de nombres réels ou de caractères non numériques).

Premier exemple d'exécution (les valeurs saisies sont en caractère gras)

```
Veillez entrer la valeur de depart : 156
Veillez entrer la valeur de pas : 0
*** Le pas doit être non nul.
Veillez entrer la valeur de pas : 2
Veillez entrer la valeur de limite : 140
*** La valeur limite doit être supérieure à la valeur de départ.
Veillez entrer la valeur de limite : 156
*** La valeur limite doit être supérieure à la valeur de départ.
Veillez entrer la valeur de limite : 171
La somme est 170 et le nombre de pas est 7
```

Deuxième exemple d'exécution (les valeurs saisies sont en caractère gras)

```
Veillez entrer la valeur de depart : 678
Veillez entrer la valeur de pas : -13
Veillez entrer la valeur de limite : 699
*** La valeur limite doit être inférieure à la valeur de départ.
Veillez entrer la valeur de limite : 500
La somme est 509 et le nombre de pas est 13
```

\*\*\* N'utilisez ni les exceptions, ni les méthodes (sauf main) pour cette question.

```
public class Question8 {

    public static void main ( String [] args ) {
        final String messInviteDepart =
            "Veuillez entrer la valeur de depart :";
        final String messInvitePas =
            "Veuillez entrer la valeur de pas :";
        final String messInviteLimite =
            "Veuillez entrer la valeur de limite :";
        final String messErreurPas =
            "*** Le pas doit être non nul.";
        final String messErreurLimiteSup =
            "*** La valeur limite doit être supérieure à la valeur de départ.";
        final String messErreurLimiteInf =
            "*** La valeur limite doit être inférieure à la valeur de départ.";

        // à compléter (veuillez utiliser les six constantes String précédentes)
    }
}
```



