

L'informatique? Algorithme



Algorithme (M. al-Khwarizmi - 780:850)

Un algorithme est une suite ordonnée d'instructions qui indique la démarche à suivre pour résoudre un problème.

Exemple: Faire cuire un oeuf sur le plat

Acheter un oeuf; mettre de l'huile dans la poele; faire chauffer; casser l'oeuf dans la poele; attendre 2 minutes.

Algorithmique

L'algorithmique est la science des algorithmes. (construction d'algorithme, vérification de la validité, la robustesse, leur réutilisabilité, la complexité et leur efficacité)



L'informatique? Programme

```
for(vv=1;vv<=rank2-1; vv++)
{ bool3=true;
  for(v=1;v<=k-1;v++)
  { if(Math.abs(t[v]-t[v+1])>1)
    {
      bool1=false;
      for(v1=1;v1<=k-1;v1++)
      if((tt[vv][v1]==t[v] && tt[vv][v1+1]==t[v+1])
        bool1=true;
      if(bool1==false)
        bool3=false;
    }
  if(Math.abs(tt[vv][v]-tt[vv][v+1])>1)
```

Programme

Il s'agit de la traduction d'un algorithme dans un langage particulier. Un programme dépend donc du langage, et également de la plateforme d'exécution.

Programmation

C'est l'action qui consiste à rédiger un programme dans un langage donné.

L'informatique? Compilation et Exécution

```
for(vv=1;vv<=rank2-1; vv++)
{ bool3=true;
  for(v=1;v<=k-1;v++)
  { if(Math.abs(t[v]-t[v+1])>1)
    {
      bool1=false;
      for(v1=1;v1<=k-1;v1++)
        if((tt[vv][v1]==t[v] &&tt[vv][v1+1]==t[v+1])
          bool1=true;
      if(bool1==false)
        bool3=false;
    }
    if(Math.abs(tt[vv][v]-tt[vv][v+1])>1)
```



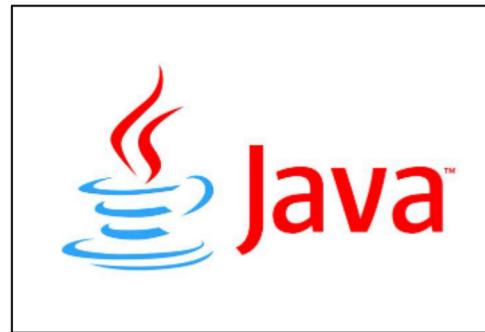
Compilation

C'est l'action qui consiste à traduire le programme en langage compréhensible par l'ordinateur.

Interpréteur (en JAVA)

Un outil ayant pour tâche d'exécuter un programme qui a été compilé.

La programmation en Java



Installation de Java

<http://www.oracle.com>

La programmation en Java

La structure d'un programme java

```
public class monpremierprogramme      // Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        //declaration de variables
        //instructions //commentaires
    }
}
```

Comment Faire?

- Dans un fichier texte, taper le code ci-dessus
- Sauvegarder le fichier texte dans un fichier
monpremierprogramme.java
- Compiler le fichier: **javac monpremierprogramme.java**
(création du fichier monpremierprogramme.class)
- Exécuter le programme: **java monpremierprogramme**



La programmation en Java

Bit (binary digit)

0 ou 1: il s'agit de la plus petite unité d'information manipulable par une machine

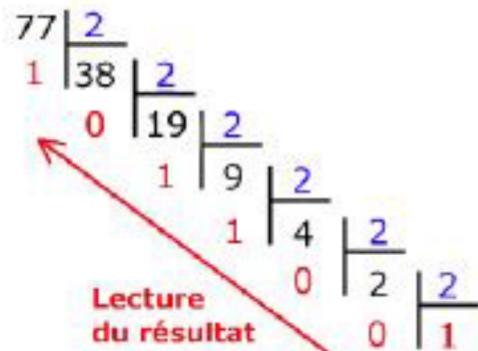
Octet (Byte en anglais)

Unité d'information composée de 8 bits

Unités standardisées

- 1 kilooctet (ko) = 1000 octets
- 1 megaoctet (Mo) = 10^6 octets
- 1 gigaoctet (Go) = 10^9 octets
- 1 tetraoctet (To) = 10^{12} octets
- 1 pétaoctet (Po) = 10^{15} octets

Représentation des entiers en binaire (division entière par 2)



1001101

La programmation en Java

Du binaire au décimal

Code binaire: **1001101**

$$77 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6$$

Avec 1, 2 ou 3 bits

- Avec un bit, on code les entiers 0 et 1
- Avec deux bits, on code les entiers 0, 1, 2 et 3
- Avec 3 bits, on code les entiers 0,1,2,3,4,5,6,7
- Avec n bits, on code les entiers compris entre 0 et $2^n - 1$

La programmation en Java

Comment coder un réels???? Exemple de 77,3

- Codage partie entière : 77 \longleftrightarrow 1001101
- Codage partie non entière : 0.3 \longleftrightarrow ????

$$0.3 * 2 = 0.6$$

$$0.6 * 2 = 1.2$$

$$0.2 * 2 = 0.4$$

$$0.4 * 2 = 0.8$$

$$0.8 * 2 = 1.6$$

$$0.6 * 2 = 1.2$$

$$\dots = \dots$$

$$0.3 \longleftrightarrow 0.0 \ 1001 \ 1001 \ 1001 \ \dots$$

Codage de 77.3

$$77.3 \longleftrightarrow 1001101.0 \ 1001 \ 1001 \ 1001 \ \dots$$



La programmation en Java

Les types de données simples en Java

Type1	Octets	Marge des valeurs
int	4	entier : -2 147 483 648 et 2 147 483 647
short	2	entier : -32 768 et 32 767
long	8	entier: -9 223 372 036 854 775 808 et 9 223 372 036 854 775 807
byte	1	entier: -128 to 127
float	4	réel jusqu'à 10^{38}
double	8	réel jusqu'à 10^{308}
char	2	Un seul caractère. (exp 'a' 'B' 'c' '&')
boolean	1	Valeur true ou false

La programmation en Java

Les variables

Une variable est un contenant ayant un nom (identificateur) et pouvant contenir une information des types précédents (int, double, char, ...)

- C'est une boite ayant un nom et pouvant contenir une valeur d'un seul type de données.
- Toute variable d'un programme doit être déclarée avant son utilisation (en début de programme par exemple)

Déclaration d'une variable

Type de données Identificateur ;

Déclaration de la variable toto pouvant contenir un entier

int toto ;

Les opérateurs

- L'addition, la soustraction, la multiplication, la division :
+, -, *, /
- La division entière et le reste de la division entière : /, %
- Les comparaisons : <, >, ==, <=, >=, !=
- Affectation : =
- Le Non : !
- Le ET logique : &&
- Le Ou (exclusif) : ^
- Le Ou (Inclusif) : ||

Les opérations classiques

Une expression mathématique:

$$\frac{2x + \frac{3y-1}{5x+1}}{4x} \xrightarrow{\text{ENJAVA}} (2 * x + (3 * y - 1) / (5 * x + 1)) / (4 * x)$$

Reste de la division entière:

$$x \bmod 10 \xrightarrow{\text{ENJAVA}} x \% 10$$

Division entière d'une variable entière a :

$$\frac{a}{2} \xrightarrow{\text{ENJAVA}} a/2$$

La programmation en Java

Affectation

C'est placer une valeur dans une variable

Comment faire:

Indiquer la valeur exacte à placer dans la variable toto de la façon suivante:

toto = valeur à placer;

Examples

titi= 8 ; // place 8 dans la variable titi

toto= titi ; // place le contenu de titi dans la variable toto

toto= titi+5 ; // place la somme du contenu de titi avec 5 dans toto

toto= toto+4 ; // place la somme du contenu de toto avec 4 dans toto

La programmation en Java

Les opérateurs booléens

Non : !

Ou exclusif : \wedge (Correspond au **OU BIEN**)

	true	false
true	false	true
false	true	false

Ou inclusif : ||

	true	false
true	true	true
false	true	false

Et : &&

	true	false
true	true	false
false	false	false



La programmation en Java

Le type boolean

Le type boolean permet uniquement deux valeurs: **true**, **false**

Exercice: Donner une expression java

$$a = b \xrightarrow{\text{ENJAVA}} a == b$$

$$a \geq b \xrightarrow{\text{ENJAVA}} a \geq b$$

$$x \in [a, b] \xrightarrow{\text{ENJAVA}} x \geq a \&\& x \leq b$$

$$x \in]a, b] \xrightarrow{\text{ENJAVA}} x > a \&\& x \leq b$$

$$x \in]a, b] \cap [c, d[\xrightarrow{\text{ENJAVA}} (x > a \&\& x \leq b) \&\& (x \geq c \&\& x < d)$$



Exercice: Donner une expression java

$x \in]a, b] \cup [c, d[\xrightarrow{\text{ENJAVA}} (x > a \&\& x \leq b) \mid\mid (x \geq c \&\& x < d)$

$[a, b] \subset [c, d] \xrightarrow{\text{ENJAVA}} a \geq c \&\& b \leq d$

x est une année bissextile lorsque x est un multiple de 4 mais pas de 100 OU BIEN x est un multiple de 400.

$\xrightarrow{\text{ENJAVA}} (x \% 4 == 0 \&\& x \% 100 != 0) \wedge (x \% 400 == 0)$

La programmation en Java

La négation d'une expression

- * négation (Je fais un CM **ET** il fait beau) \leftrightarrow (je ne fais pas un CM **OU** il ne fait pas beau)
- * $!(A \&\& B) \leftrightarrow !A \parallel !B$
- * négation (Je fais un CM **OU** je suis dans mon bureau) \leftrightarrow (je ne fais pas un CM **ET** je ne suis pas dans mon bureau)
- * $!(A \parallel B) \leftrightarrow !A \&\& !B$

Donner l'expression de la négation

On a vu que:

$$x \in]a, b] \cup [c, d[\stackrel{\text{ENJAVA}}{\implies} (x > a \&\& x \leq b) \parallel (x \geq c \&\& x < d)$$

$$x \notin]a, b] \cup [c, d[\stackrel{\text{ENJAVA}}{\implies} (x \leq a \parallel x \geq b) \&\& (x < c \parallel x \geq d)$$

La programmation en Java

```
public class monpremierprogramme //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        // Déclaration des variables
        int toto, titi, tutu; // Déclare trois variables
        boolean tyty; // Déclare une variable booléenne

        // Instructions
        toto= 4; // Met la valeur 4 dans toto
        toto= toto+1; // Ajoute 1 au contenu de toto
        titi= toto*4; // Met 4*5 dans titi

        tutu= titi/3; // Met 20/3 dans tutu
        tutu= titi % 3; // Met 20 % 3 dans tutu

        tyty= false; // Met false dans tyty
        tyty= tutu>=2 && tutu < 4; // Met true dans tyty
    }
}
```

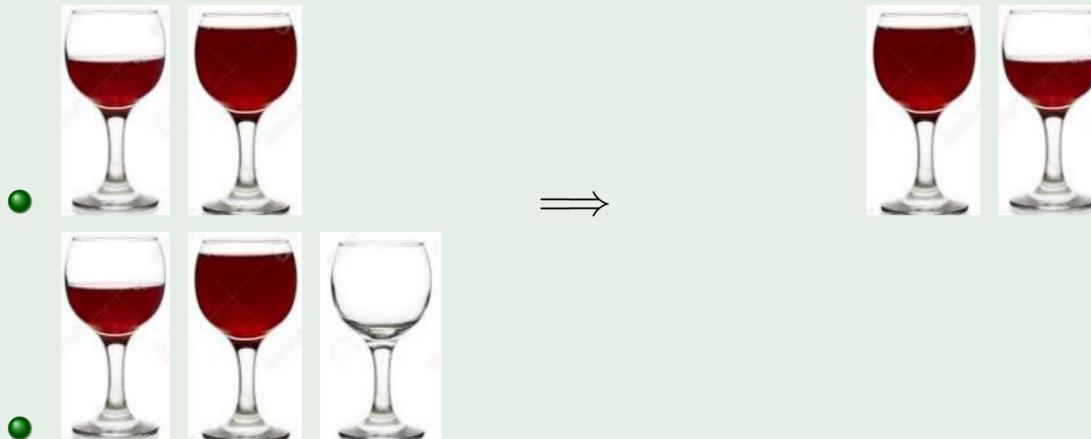
La programmation en Java

Echange du contenu de deux variables:



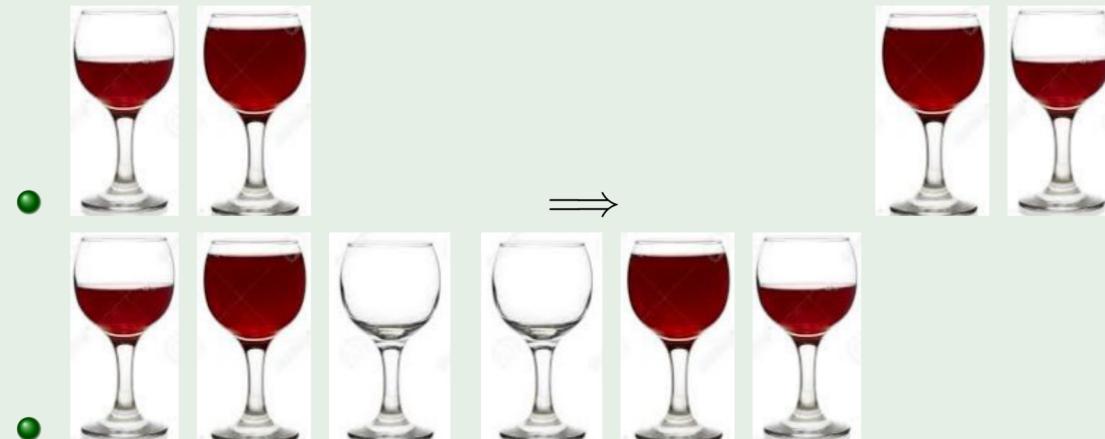
La programmation en Java

Echange du contenu de deux variables:



La programmation en Java

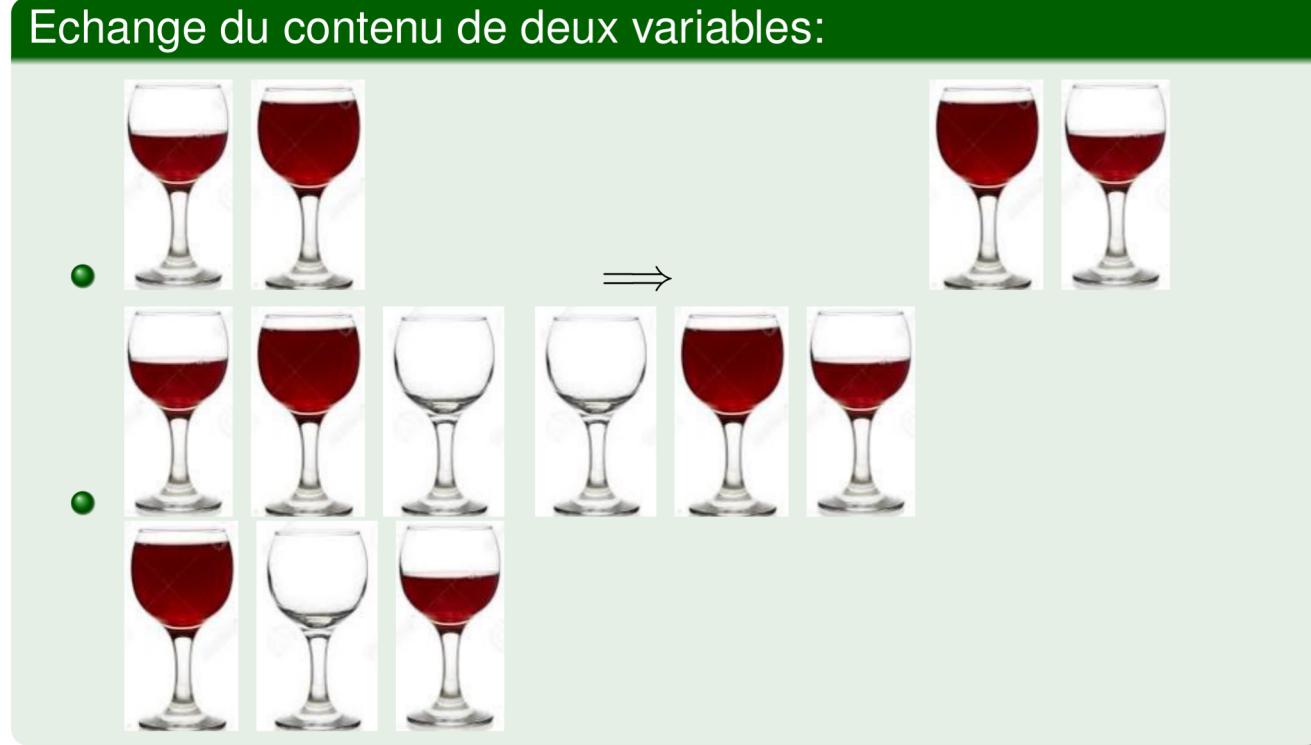
Echange du contenu de deux variables:



Etape 1: $V = N$

La programmation en Java

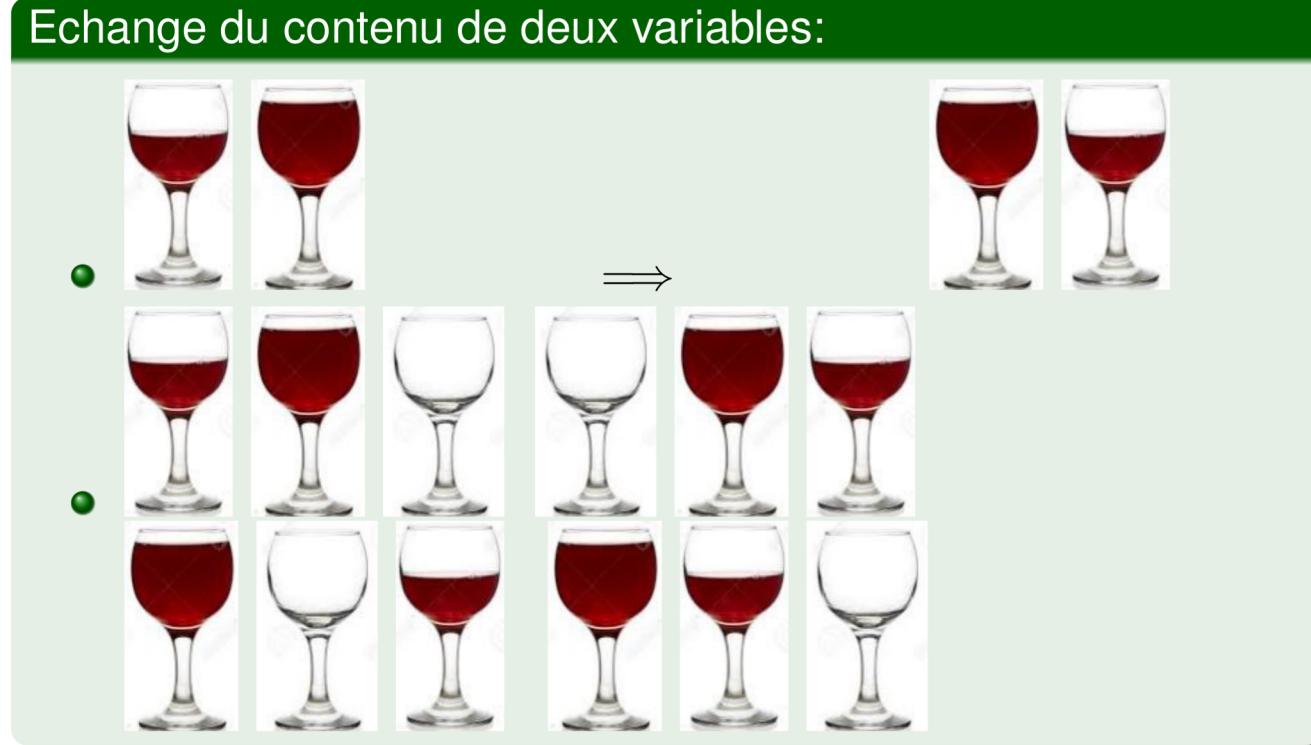
Echange du contenu de deux variables:



Etape 1: $V = N$ Etape 2: $N = G$

La programmation en Java

Echange du contenu de deux variables:



Etape 1: $V = N$ Etape 2: $N = G$ Etape 3: $G = V$

La programmation en Java

```
public class echange //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        // Déclaration des variables
        int Nuits, Gevrey, Temp ; // Déclare trois variables

        // Remplissage des verres
        Nuits=4;           // Remplit le verre de Nuits
        Gevrey=8;          // Remplit le verre de Gevrey
        // Echange le contenu des deux verres
        Temp=Nuits;        // Verse le contenu de Nuits dans Temp
        Nuits=Gevrey;      // Verse le contenu de Gevrey dans Nuits
        Gevrey=Temp;        // Verse le contenu de Temp dans Gevrey
    }
}
```

La programmation en Java



Communiquer avec l'ordinateur pendant l'exécution du programme

- Affichage des résultats à l'écran (sortie)
- Saisie d'informations au clavier (entrée)

La programmation en Java - Les sorties

Affichage à l'écran d'un texte

```
System.out.print("texte_a_afficher");
// Affiche le texte

System.out.println("texte_a_afficher");
// Affiche le texte puis revient à la ligne
```

Affichage à l'écran du contenu d'une variable

```
System.out.print(variable);
// Affiche le contenu de la variable

System.out.println(variable);
// Affiche le contenu de la variable puis revient à la ligne
```

La programmation en Java - Les sorties

Affichage à l'écran d'un texte et du contenu d'une variable

```
System.out.print("texte_a_afficher"+variable);
// Affiche le texte puis le contenu de la variable

System.out.println("texte_a_afficher"+variable);
// Affiche le texte puis le contenu de la variable et
// revient à la ligne

System.out.println("toto=_"+toto);
//Affiche toto= 8

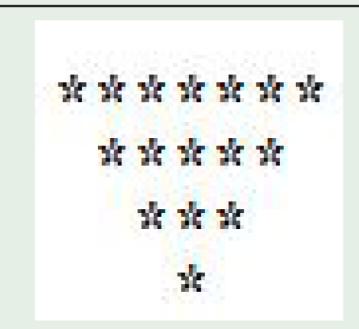
System.out.println("toto=_"+toto+"_et_titi=_"+titi);
// Affiche toto= 8 et titi= 4
```

La programmation en Java - Les sorties

```
public class echange //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        int Nuits, Gevrey, Temp ;
        System.out.println("Bonjour");
        Nuits=4;
        Gevrey=8;
        Temp=Nuits;
        Nuits=Gevrey;
        Gevrey=Temp;
        System.out.println("Le contenu de Nuits est :" +Nuits);
        System.out.println("Le contenu de Gevrey est :" +Gevrey);
    }
}
```

La programmation en Java - Les sorties

Un joli dessin!!!



```
public class jolidessin //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        System.out.println("*****");
        System.out.println("_****_");
        System.out.println("_**_*_");
        System.out.println("_*_*_*_*");
    }
}
```

La programmation en Java - Les entrées

La saisie d'un entier dans la variable toto

```
int toto;  
toto=Lire.i();  
// attend que l'utilisateur saisisse un entier
```

La saisie d'un réel dans la variable titi

```
double titi;  
titi=Lire.d();  
// attend que l'utilisateur saisisse un réel
```

La saisie d'un texte dans une variable tutu

```
String tutu;  
tutu=Lire.S();  
// attend que l'utilisateur saisisse un texte
```



La programmation en Java - Les Entrées/Sorties

```
public class echange //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        int Nuits, Gevrey, Temp ;
        System.out.println("Saisir le contenu de Nuits");
        Nuits=Lire.i();
        System.out.println("Saisir le contenu de Gevrey");
        Gevrey=Lire.i();
        Temp=Nuits;
        Nuits=Gevrey;
        Gevrey=Temp;
        System.out.println("Le contenu de Nuits est:"+Nuits);
        System.out.println("Le contenu de Gevrey est:"+Gevrey);
    }
}
```

Exercice - Les Entrées/Sorties

Moyenne de deux réels

Ecrire un programme Java permettant de faire la moyenne de deux réels saisis par l'utilisateur.

```
public class moyenne //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        double x, y, moyenne;
        System.out.println("Saisir_x");
        x=Lire.d();
        System.out.println("Saisir_y");
        y=Lire.d();
        moyenne=(x+y)/2;
        System.out.println("La_moyenne_est:"+moyenne);
    }
}
```

Fiche Memo CM1- Les Entrées/Sorties

Le minimum à savoir!!!!

```
public class nomprog //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        // Déclaration des variables utiles
    }
}
```

Le minimum à savoir!!!!

- ① ; à la fin de chaque instruction
- ② Affectation: x=3;
- ③ Opérateurs booléen: &&, ||, ==,
 $x \geq 1 \ \&\amp; \ x \leq 5,$ $x == 7 \ || \ y == 8$
- ④ Affichage à l'écran: System.out.println("Bonjour"+variable);
- ⑤ Saisie au clavier: x=Lire.i(); y=Lire.d();



Exercice - Les Entrées/Sorties

Résolution de $a * x = b$ lorsque $a \neq 0$

Ecrire un programme Java permettant la saisie de a et b par l'utilisateur, puis d'afficher la solution de l'équation.

```
public class moyenne //Nom du programme
{
    public static void main (String[] args) //Programme principal
    {
        // Déclaration des variables utiles

        // Saisie de a et b

        // Calcul de la solution

        // Affichage de la solution
    }
}
```