

DUT MMI – IUT de Marne-la-Vallée
13/11/2013
M1202 - Algorithmique

Cours 4

Les fonctions

Sources

- *Le livre de Java premier langage*, d'A. Tasso
- Cours INF120 de J.-G. Luque
- Cours FLIN102 de l'Université Montpellier 2
- Cours de J. Henriet : <http://julienhenriet.olymp-network.com/Algo.html>

Plan du cours 4 – Les fonctions

- Résumé de l'épisode précédent
- Les fonctions, entrées et sortie
- Combiner plusieurs fonctions pour en créer une nouvelle
- Écriture de fonctions en Java, variables locales

Plan du cours 4 – Les fonctions

- Résumé de l'épisode précédent
- Les fonctions, entrées et sortie
- Combiner plusieurs fonctions pour en créer une nouvelle
- Écriture de fonctions en Java, variables locales

Résumé de l'épisode précédent

Tableaux :

- pour stocker un ensemble de valeurs **de même type**
- **une** valeur par case
- **nombre de cases fixé** à l'initialisation du tableau
- boucle pour **parcourir le tableau**

Boucles :

- boucle **Tant que** et boucle **Pour tout**
- attention à l'**initialisation** et la **condition d'arrêt** (premier et dernier passage dans la boucle)

Plan du cours 4 – Les fonctions

- Résumé de l'épisode précédent
- Les fonctions, entrées et sortie
- Combiner plusieurs fonctions pour en créer une nouvelle
- Écriture de fonctions en Java, variables locales

Les fonctions

La "minute mathématique"

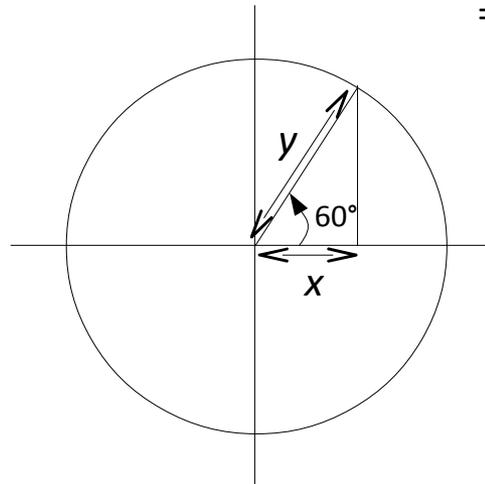
fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$		
somme			
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$		
somme			
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

$$\begin{aligned}\text{cosinus}(1.047) &\approx \text{cosinus}(\pi/3) = \text{cosinus}(60^\circ) \\ &= x/y \\ &= 0.5/1\end{aligned}$$



Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme			
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers	entier
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	cosinus(1.047)=0.5	flottant	flottant
somme	somme(2,3)=5	2 entiers	entier
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les "nombres à virgule" se notent avec un point aux Etats-Unis

La virgule sépare les paramètres d'une fonction

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé			
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse			
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence			
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant
estPositif			
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant
estPositif	$\text{estPositif}(-5)=\text{FAUX}$	entier	booléen
partieEntière			
moyenne			
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant
estPositif	$\text{estPositif}(-5)=\text{FAUX}$	entier	booléen
partieEntière	$\text{partieEntière}(5.6)=5$	flottant	entier
moyenne			
min			

$\text{PartieEntière}(10)=\lfloor 10 \rfloor = 10$

$\text{PartieEntière}(5.6)=\lfloor 5.2 \rfloor = 5$

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant
estPositif	$\text{estPositif}(-5)=\text{FAUX}$	entier	booléen
partieEntière	$\text{partieEntière}(5.6)=5$	flottant	entier
moyenne	$\text{moyenne}(2,4,6)=4$	3 flottants	flottant
min			

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
cosinus	$\text{cosinus}(1.047)=0.5$	flottant	flottant
somme	$\text{somme}(2,3)=5$	2 entiers 2 flottants	entier flottant
opposé	$\text{opposé}(4)=-4$	entier flottant	entier flottant
inverse	$\text{inverse}(10)=0.1$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant
estPositif	$\text{estPositif}(-5)=\text{FAUX}$	entier	booléen
partieEntière	$\text{partieEntière}(5.6)=5$	flottant	entier
moyenne	$\text{moyenne}(2,4,6)=4$	3 flottants	flottant
min	$\text{min}(\{6,2,4,3\})=2$	tableau d'entiers tableau de flottants	entier flottant

Plan du cours 4 – Les fonctions

- Résumé de l'épisode précédent
- Les fonctions, entrées et sortie
- Combiner plusieurs fonctions pour en créer une nouvelle
- Écriture de fonctions en Java, variables locales

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
somme	$\text{somme}(2,3)=5$	flottant entier	flottant entier
opposé	$\text{opposé}(4)=-4$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant

Fonction différence
Entrées : 2 entiers a et b
Sortie : entier
Début
 Renvoyer ...
Fin

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
somme	<code>somme(2,3)=5</code>	flottant entier	flottant entier
opposé	<code>opposé(4)=-4</code>	flottant	flottant
différence	<code>différence(2,3)=-1</code>	2 entiers 2 flottants	entier flottant

Fonction **différence**

Entrées : 2 entiers a et b

Sortie : entier

Début

Renvoyer `somme(a,opposé(b))`

Fin

Attention, si on utilise la fonction somme qui renvoie un flottant, le type de sortie n'est pas correct pour la fonction différence !

Les fonctions

La "minute mathématique"

fonction	exemple	entrées possibles	sortie
somme	$\text{somme}(2,3)=5$	flottant entier	flottant entier
opposé	$\text{opposé}(4)=-4$	flottant	flottant
différence	$\text{différence}(2,3)=-1$	2 entiers 2 flottants	entier flottant

Fonction **différence**

Entrées : 2 flottants a et b

Sortie : flottant

Début

 Renvoyer **somme**(a ,**opposé**(b))

Fin

Les fonctions

Les **fonctions** = les **algorithmes**

Un ou plusieurs paramètres en entrée...

Une sortie...

Les fonctions

Les **fonctions** = les **algorithmes**

Un ou plusieurs paramètres en entrée...
ou aucun

Une sortie...
ou aucune

Plan du cours 4 – Les fonctions

- Résumé de l'épisode précédent
- Les fonctions, entrées et sortie
- Combiner plusieurs fonctions pour en créer une nouvelle
- Écriture de fonctions en Java, variables locales

Les fonctions en Java – La fonction `main`

Toujours une fonction `main` qui ne renvoie rien

Déclaration des fonctions après la fonction `main`

Les fonctions en Java – La fonction `main`

Toujours une fonction `main` qui ne renvoie rien et qui prend en entrée les paramètres du programme

Déclaration des fonctions après la fonction `main`

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

Les fonctions en Java – La fonction main

Toujours une fonction `main` qui ne renvoie rien
et qui prend en entrée les **paramètres du programme**

Déclaration des fonctions après la fonction `main`

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1
j="+j+", somme : "+ad
    }
    public static int inc(in
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

En ligne de commande :

```
java TP1 toto 1 10.5
```

Le tableau `arg` est alors :

```
{"toto", "1", "10.5"}
```

Les fonctions en Java – La fonction `main`

Toujours une fonction `main` **qui ne renvoie rien**
et qui **prend en entrée les paramètres du programme**

Déclaration des fonctions après la fonction `main`

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

Les fonctions en Java – Autres fonctions

Toujours une fonction `main` **qui ne renvoie rien**
et qui **prend en entrée les paramètres du programme**

Déclaration des fonctions après la fonction `main`

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
        j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

type de variable renvoyée par la fonction



Les fonctions en Java – Autres fonctions

Toujours une fonction `main` **qui ne renvoie rien**
et qui **prend en entrée les paramètres du programme**

Déclaration des fonctions après la fonction `main`

```
public class TP1{  
    public static void main(String[] arg) {  
        int i,j;  
        i=5;  
        j=34;  
        System.out.print("i+1="+inc(i)+"", i="+i+",  
        j="+j+", somme : "+addition(i,j));  
    }  
    public static int inc(int i){  
        i=i+1;  
        return i;  
    }  
    public static int addition(int i, int j){  
        return i+j;  
    }  
}
```

type de variable renvoyée par la fonction

**entrées de la fonction
précédées de leur type**

Les fonctions en Java – Autres fonctions

Toujours une fonction `main` **qui ne renvoie rien**
et qui **prend en entrée les paramètres du programme**

Déclaration des fonctions après la fonction `main`

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
        j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i) {
        i=i+1;
        return i;
    }
    public static int addition(int i, int j) {
        return i+j;
    }
}
```

 **appel de la fonction addition**

Les fonctions en Java – Autres fonctions

Seule **la fonction main** s'exécute quand on exécute le programme TP1.

Les **autres fonctions** ne s'exécutent que si elles sont **appelées** pendant l'exécution de la fonction `main`

```
public class TP1{
    public static void main(String[] arg){
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

appel de la fonction inc

appel de la fonction addition

Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg){
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
        j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg) {
        int i, j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
        j="+j+" , somme : "+addition(i, j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

**pas la même variable *i*
même si elles ont la
même valeur !**



Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

définition de la fonction addition

Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

déclaration de la fonction addition

définition de la fonction addition

Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
    }
    public static int inc(int i){
        i=i+1;
        return i;
    }
    public static int addition(int i, int j){
        return i+j;
    }
}
```

déclaration de la fonction addition

renvoi du résultat en sortie

définition de la fonction addition

Les fonctions en Java

Visibilité des variables : toute variable déclarée à l'intérieur d'une fonction n'est valable **que dans cette fonction** et **ne peut pas être utilisée ailleurs**.

Variables locales

```
public class TP1{
    public static void main(String[] arg) {
        int i,j;
        i=5;
        j=34;
        System.out.print("i+1="+inc(i)+" , i="+i+" ,
            j="+j+" , somme : "+addition(i,j));
        }
        appel de la fonction addition
    public static int inc(int i){
        i=i+1;
        return i;
        }
        déclaration de la fonction addition
    public static int addition(int i, int j) {
        return i+j; renvoi du résultat en sortie
    }
}
définition de la fonction addition
```