Variable ...
Vous avez dit variable ...



Pascal Lafourcade

UNIVERSITÉ Clermont Auvergne

LIMOS

IREM Institut de Recherche sur l'Enseignement des Mathématiques Clermont-Ferrand

LIMOS
LABORATOIRE D'INFORMATIQUE,
DE MODÉLISATION ET D'OPTIMISATION DES SYSTÈMES

15 Janvier 2019

Définition du dictionnaire

Variable

- Qui est sujet à variations, qui peut changer souvent au cours d'une durée, selon les circonstances.
- Qui peut prendre plusieurs valeurs distinctes selon les circonstances.
- Qui est conçu, fabriqué de façon à pouvoir subir des variations.

Définition du dictionnaire

Variable

- Qui est sujet à variations, qui peut changer souvent au cours d'une durée, selon les circonstances.
- Qui peut prendre plusieurs valeurs distinctes selon les circonstances.
- Qui est conçu, fabriqué de façon à pouvoir subir des variations.

MATH., INFORMAT., PHYS.

Symbole, terme, phénomène observable auquel différentes valeurs prises dans un ensemble peuvent être attribuées.

Définition du dictionnaire

Variable

- Qui est sujet à variations, qui peut changer souvent au cours d'une durée, selon les circonstances.
- Qui peut prendre plusieurs valeurs distinctes selon les circonstances.
- Qui est conçu, fabriqué de façon à pouvoir subir des variations.

MATH., INFORMAT., PHYS.

Symbole, terme, phénomène observable auquel différentes valeurs prises dans un ensemble peuvent être attribuées.

Antonyme : constante

Vous avez dit variable ...

$A = 2\sin x + 3x$

$y = ax + b$

$\forall x \exists y, x + 3 > y$

$f(x) = sin(x)$

$$E[X] := \sum_{u \in U} X(u)\Delta P[u]$$

$x = x + 1$

$if\ x > 0\ then$

```
def f(x) :
    return x+1
```

$4x^2 - 2x + 1 = 0$

L'implicite ...

En mathématiques

$$A = 2\sin x + 3x$$

L'implicite ...

En mathématiques

$$A = 2\sin x + 3x$$

La valeur de A *dépend* de la valeur de x.

L'implicite ...

En mathématiques

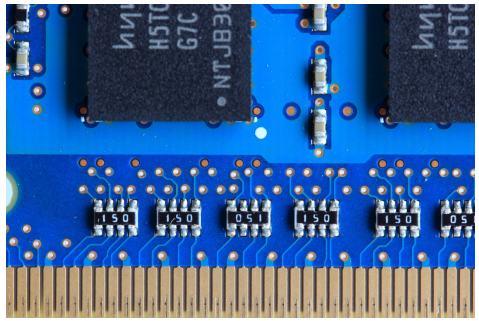
$$A = 2\sin x + 3x$$

La valeur de A *dépend* de la valeur de x.

- x prend ses valeurs dans un ensemble donné D.
- À *chaque* valeur de x correspond une valeur de A.
- il faudrait écrire :

$$\forall x \in D, A(x) = 2\sin x + 3x$$

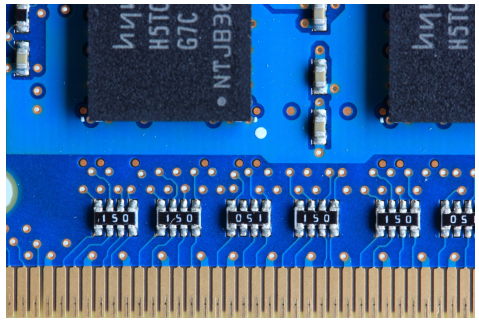
En informatique



Notion de variable

- Une variable désigne un **emplacement** de la **mémoire** qui porte un **nom**.

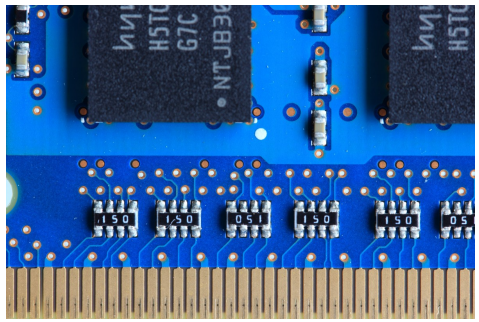
En informatique



Notion de variable

- Une variable désigne un **emplacement** de la **mémoire** qui porte un **nom**.
- Contient une **valeur**.

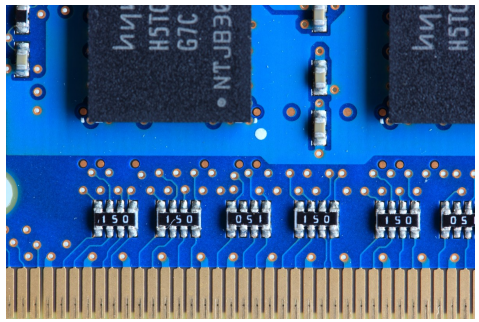
En informatique



Notion de variable

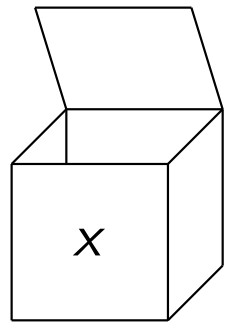
- Une variable désigne un **emplacement** de la **mémoire** qui porte un **nom**.
- Contient une **valeur**.
- Son contenu peut changer durant l'exécution d'un programme.

En informatique



Notion de variable

- Une variable désigne un **emplacement** de la **mémoire** qui porte un **nom**.
- Contient une **valeur**.
- Son contenu peut changer durant l'exécution d'un programme.





Plan

L'état et les types

Égalité

Variable globale vs locale

Passage des variables

Conclusion

Plan

L'état et les types

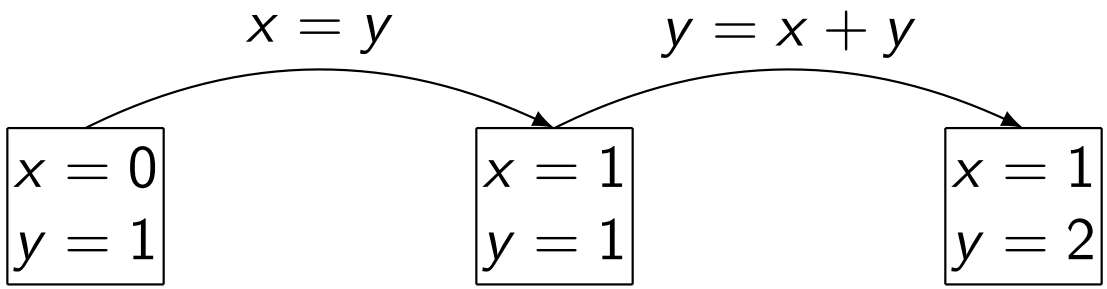
Égalité

Variable globale vs locale

Passage des variables

Conclusion

La notion d'état



- L'ensemble des variables présentes en mémoire
- et leurs valeurs respectives

constitue l'*état* de la mémoire.

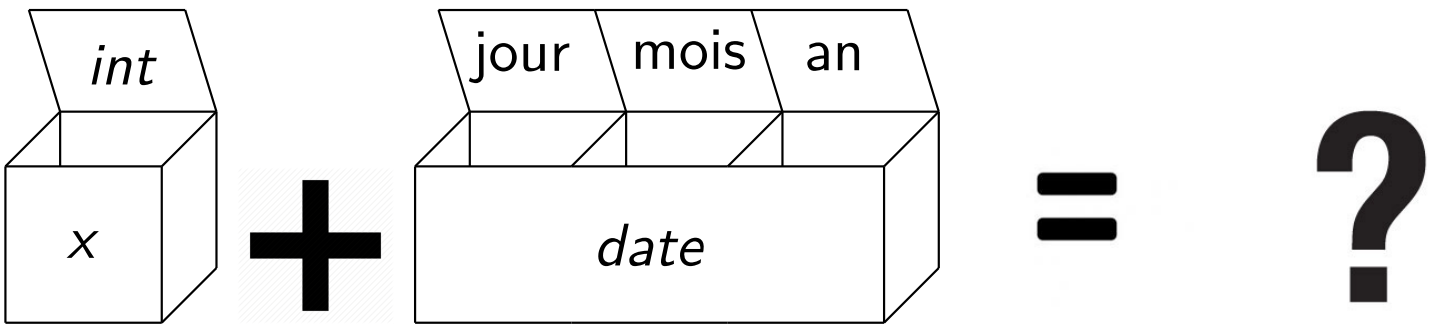
Une instruction d'un programme ne « fait » pas quelque chose, elle « fait quelque chose à l'état ».

Des choux et des carottes



Des choux et des carottes





Taille des boîtes prédéfinies ...

... espace mémoire fini (2^{64}, tables vs listes chaînées).

Type des fonctions

Des fonctions

- $f(x) = x^2 : \mathbb{R} \to \mathbb{R}$ ou `float` $\to$ `float` ou `int` $\to$ `int`

Type des fonctions

Des fonctions

- $f(x) = x^2 : \mathbb{R} \to \mathbb{R}$ ou `float` $\to$ `float` ou `int` $\to$ `int`
- `estplusgrand(a,b)` : `int * int` $\to$ `bool`
  ou `float * float` $\to$ `bool`
  ou `char * char` $\to$ `bool`

Type polymorphe α

Type de la fonction estegale(x,y)

`int * int` $\rightarrow$ `bool`

`float * float` $\rightarrow$ `bool`

`char * char` $\rightarrow$ `bool`

Type polymorphe α

Type de la fonction estegale(x,y)

`int * int` $\rightarrow$ `bool`

`float * float` $\rightarrow$ `bool`

`char * char` $\rightarrow$ `bool`

α * α $\rightarrow$ `bool`

Typage

Statique (en C)

```
int a ;
a=5 ;
```

Dynamique (en Python)

```
def identite(x) :
    return x
```

Conversions de type

Explicites (cast)

```
    int a;
    a=(int)3.14;
    print("a=%d",a);
a=3
```

Implicites

```
    int  i = 17;
    char c = 'c'; /* ascii value is 99 */
    float sum;
    sum = i + c;
    printf("Value of sum : %f\n", sum );
Value of sum : 116.000000
```

Plan

L'état et les types

Égalité

Variable globale vs locale

Passage des variables

Conclusion

En mathématiques :

3 égalités différentes

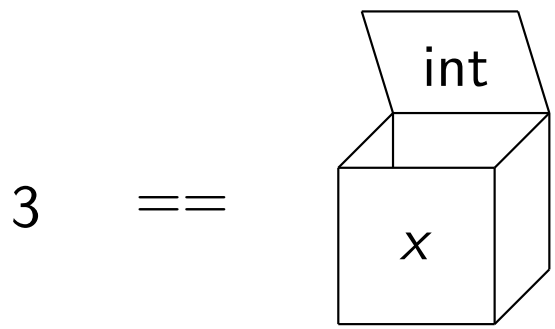
- $\forall x \in \mathbb{R}, f(x) = 2x + 1$ (égalité de définition)
- $\forall x \in \mathbb{R}, 1 = sin(x)^2 + cos(x)^2$ (propriété)
- $\exists x \in \mathbb{Z}, 2x + 1 = 0$? (équation)

En informatique :

Une autre notion d'égalité

- Comparaison des valeurs dans un état de la mémoire.
  $== : \alpha * \alpha \rightarrow bool$

Exemple : $3 == x$



Dans certains langages, le test d'égalité est noté $x = 3$.

Plan

L'état et les types

Égalité

Variable globale vs locale

Passage des variables

Conclusion

Un exemple

```
def f(z):
    x=z+a
    return x
x=3
a=3
print (f(x))
a=4
print (f(x))
```

Un exemple

```
def f(z):
    x=z+a
    return x
x=3
a=3
print (f(x))
a=4
print (f(x))
```

6 puis 7

Globale vs locale

a est une variable globale

z est une variable locale

Que fait ce programme ?

```
def f(z):
    y=x+z
    x=42
    return y

x=18
print f(x);
```

Que fait ce programme ?

```
def f(z):
    y=x+z
    x=42
    return y

x=18
print f(x);
```

y=x+z UnboundLocalError : local variable 'x' referenced before assignment

Le x=42 dit que x est locale, mais elle existe déjà

Plan

L'état et les types

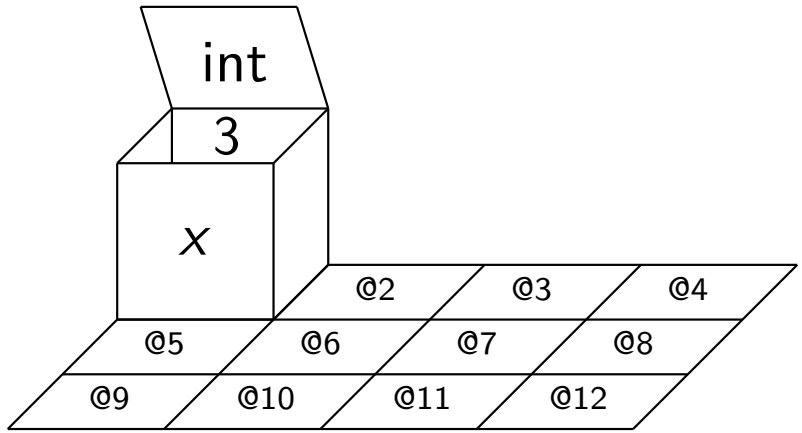
Égalité

Variable globale vs locale

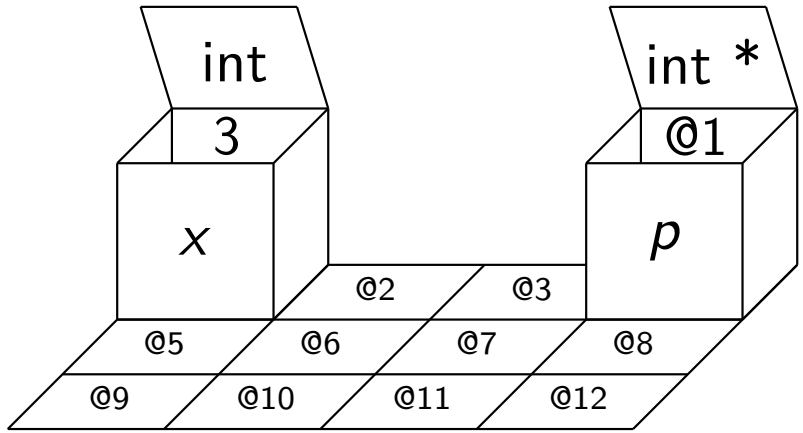
Passage des variables

Conclusion

Variables en mémoire



Références



En C

```
int i, j;
i=3;
j=i;
```

objet	adresse	valeur
i	456000	3
j	456004	3

Pointeur en C

```
int i =3;
int *p;
p=&i;
print("*p = %d \n", *p) \\ affiche 3
*p=0;
print(i = %d \n", i) \\ affiche 0
```

objet	adresse	valeur
i	456000	3
p	456004	456000
*p	456000	3

en Ocaml

Référence

```
let x = ref 0;;
val x : int ref = {contents = 0}
x := 1;;
x;;
- : int ref = {contents = 1}
```

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value		
Name		
Lazy		

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	
Name		
Lazy		

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	h(0) = 0 et r=1
Name		
Lazy		

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	h(0) = 0 et r=1
Name	f(g 0) = 1 et r=0	
Lazy		

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	h(0) = 0 et r=1
Name	f(g 0) = 1 et r=0	(g 0) + (g 0) = 0+0=0 et r=2
Lazy		

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	h(0) = 0 et r=1
Name	f(g 0) = 1 et r=0	(g 0) + (g 0) = 0+0=0 et r=2
Lazy	f(g 0) = 1 et r =0	

Lazy, name or value Evaluation

Fonctions

```
let f x = 1 ; ;
let h x = x+x ; ;
let r = ref 0 ; ;
let g x = r := !r+1 ;x ; ;
```

f g 0 vs h g 0

	f (g 0)	h (g 0)
Value	f(0) =1 et r=1	h(0) = 0 et r=1
Name	f(g 0) = 1 et r=0	(g 0) + (g 0) = 0+0=0 et r=2
Lazy	f(g 0) = 1 et r =0	(g 0) + (g 0) = 0+0=0 et r=1

Plan

L'état et les types

Égalité

Variable globale vs locale

Passage des variables

Conclusion

Conclusion

la notion de variable

- Notion clef en informatique
- État et affectation
- Type
- Liée à la notion de fonction :
  - locale vs globale
  - passage par paramétre ou par valeur

Merci pour votre attention

