

Bloc, variable, instruction

Bloc:

int main() point d'entrée (i-instruction exécutée)
{ ... } bloc

Variable:

... Variable mathématique ...

Symbole, parfois indexé, représentant une quantité inconnue appartenant à un ensemble donné.

- > généralisation
- > existence
- > résolution

... Variable informatique ...

→ mémoire du prog

associe un nom à une valeur appartenant à un ensemble donné

- > une variable correspond tjrs à une valeur

> une variable peut subir une opération.

> la valeur d'une variable peut

↳ être modifiée par une opér^{ation}

↳ la variable correspond à un emplacement mémoire et est typée.

4. types primitif :

> char (%c).

• caractère

> Int (%d)

• entier

• long : 8 byte (%ld) ou (%lu).

• Short : 2 byte (%hd ou %hu).

• unsigned : non signé (%u).

> float : (%f)

• nbr décimales (7 chiffres après)

> double (%lf)

• nbr décimales + grand. (16 chiffres)

• long double

• 16 bytes

Pointeurs:

→ ... Pointeur ... →

Variable dont la valeur est une adresse.

↳ float* pointeur

- possible de joindre des pointeurs de pointeur

Expression:

- opérateur d'opérateur + important

> stocke des valeurs en mémoire

stockage
var = exp
opérateur valeur
 opérande

↳ opérateur la valeur à droite = expression
la valeur à gauche = tire une variable

• type de la variable et de l'expression
montrent les m

• arithmétiques:

> addition: +

> soustraction: -

> division: /

> multiplication: *

• modulo: %

Comparaisons:

> prend une valeur booléenne

$\neq 0 \rightarrow \text{vrai}$

$= 0 \rightarrow \text{faux}$

> + petit : <

> + grand : >

> + petit ou = : <=

> + grand ou = : >=

> égale : ==

> ≠ : !=

Booleens:

• opérateur logique.

• et "lazy" : &&

• ou "lazy" : ||

• négatif : !

• Notation syntaxique = notation d'écriture

> Not a = exp

$a \in \{+, -, *, /, \% \}$

> ++ i ou i++ (on doit avoir -)
incrémenter littéralement.

bit à bit:

- opérat° les jours sur chacun bits de 2 opérandes
- et : &
- ou : |
- ou exclusif : ^

9 & 12

1001

1100

1000 = 8

9 | 12

1001

1100

1101 = 13

9 ^ 12

1001

1100

0101 = 5

opérateur de décalage:

- décale de x bits vers G/D.

> Ceci n'est pas une rotation

- décalage à G : <<

- décalage à D : >>

.. on conserve le signe.

- référencement : & (obtenir l'adresse).

- adressage : * (pointeur).

Priorité des opérateurs:

() [] . →	→
! -- ++ - * &	←
* / %	→
+ -	→
<< >>	→
< <= > >=	→
= = ! =	→
&	→
^	→
	→
&&	→
= += -= ...	→

Instruction simple:

- déclaration de variable
- printf / scanf
- utilisat° d'expressions dans des instruct° (manipulation de variables).
- In (tableau ligne) . It (tableau horizontal).
- Iv (tableau vertical).