

Chapitre Données & Opérations simples

Algorithmique et **Pascal**

adrien.basse@uadb.edu.sn

a solution de l'équation
pi somme
longueur du rectangle
coté du carré

La plupart des algorithmes manipulent un certain nombre de données

• entrées, sorties, intermédiaires

prix ticket déjeuner
b surface du rectangle
nombre de tickets
moyenne de l'EC
montant

**Les données d'entrée sont
transmises à l'algorithme...**

Les données de sortie sont transmises par l'algorithme...

Les données intermédiaires sont celles connues avant d'entamer l'algorithme ou calculables à partir des données d'entrée.

Lecture des entrées

Ecriture des sorties

**Affectation ou calcul des
intermédiaires**

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du périmètre d'un rectangle.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du périmètre d'un cercle.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du périmètre d'un rectangle à partir de la largeur et de la surface.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du périmètre d'un cercle à partir de son diamètre.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul des racines d'une équation du second degré.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du signe de delta d'une équation du second degré.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul de la somme de trois nombres entiers.

Entrées, sorties, intermédiaires?

Proposer un algorithme de calcul du signe du produit de trois nombres entiers.

Entrées, sorties, intermédiaires?



```
graph TD; A[Une donnée] --- B[Catégorie]; A --- C[Nom]; A --- D[Type];
```

Une donnée

Catégorie

Nom

Type

Pour chaque donnée vous devez préciser au début sa catégorie, son nom, et son type.

```
graph TD; A[Catégorie] --> B[Variable]; A --> C[Constante];
```

Catégorie

Variable

Constante

Une donnée est **constante** si

- sa **valeur** est connue avant d'entamer l'algorithme ;
- cette **valeur** ne change pas jusqu'à la fin.

Une donnée est **variable si elle n'est pas constante.**

a

solution de l'équation

pi

somme

longueur du rectangle

coté du carré

Identifier les constantes et les variables dans les algorithmes déjà écrits.

prix ticket déjeuner

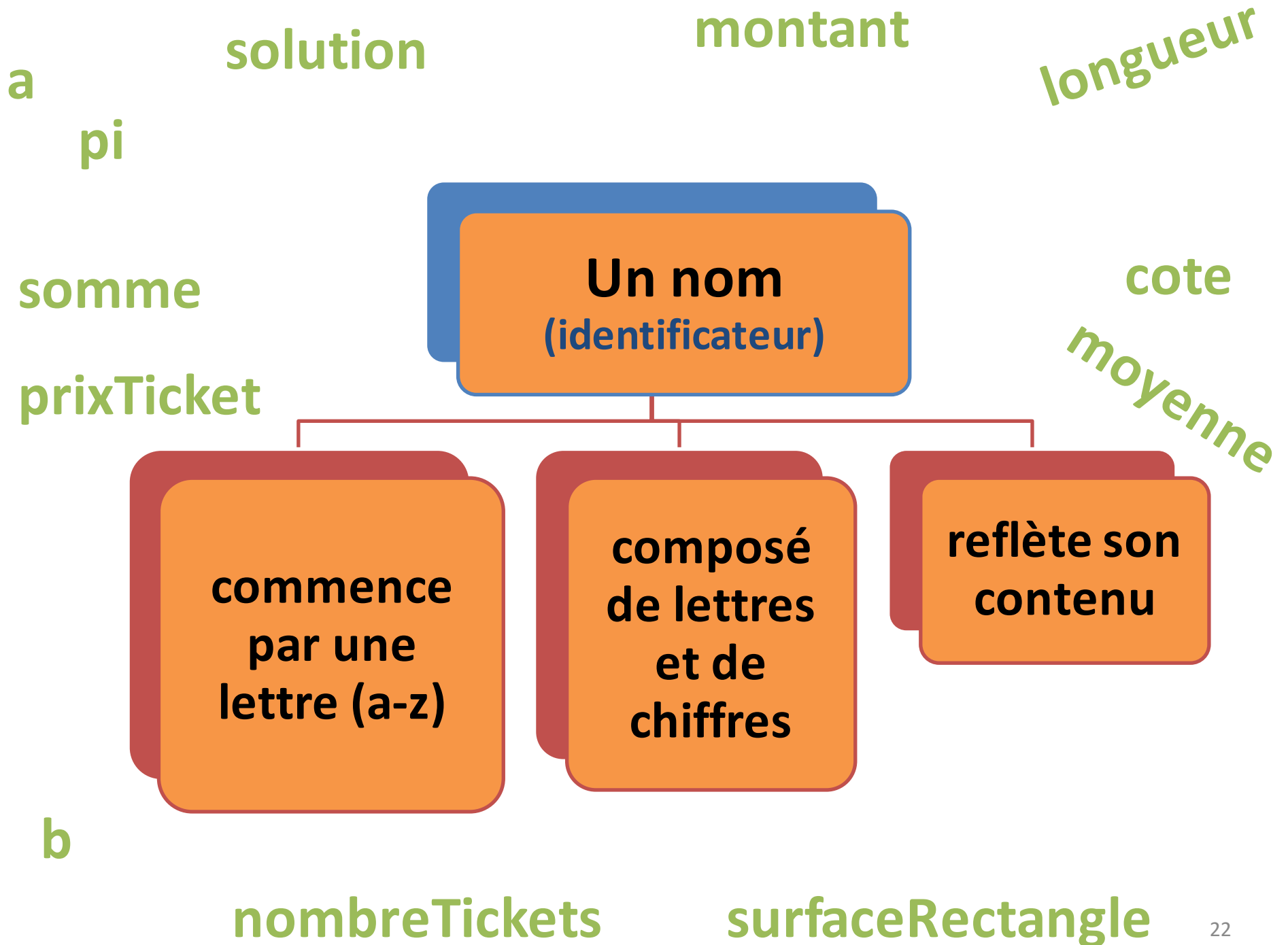
b

surface du rectangle

nombre de tickets

moyenne de l'EC

montant



a solution montant longueur
pi
1devoir
somme cote
prixTicket côté
surface rectangle
devoir1
maçon
surfaceRectangle
b
nombreTickets

Quelques conventions sur les noms.

Le nom simple d'une donnée variable commence par une lettre minuscule suivie de lettres minuscules.

largeur, perimetre.

Le nom composé d'une donnée variable comporte des mots séparés soit par:

une majuscule

perimetreCercle

Le caractère _

perimetre_cercle

Le nom simple d'une donnée constante est composée de lettres majuscules.

PRIX

**Le nom composé d'une donnée
constante comporte des mots
séparés par _.**

PRIX_UNITE

a

solution de l'équation

pi

somme

longueur du rectangle

coté du carré

Proposez des noms **corrects** en respectant les conventions présentées.

prix ticket déjeuner

b

surface du rectangle

nombre de tickets

moyenne de l'EC

montant

Un type

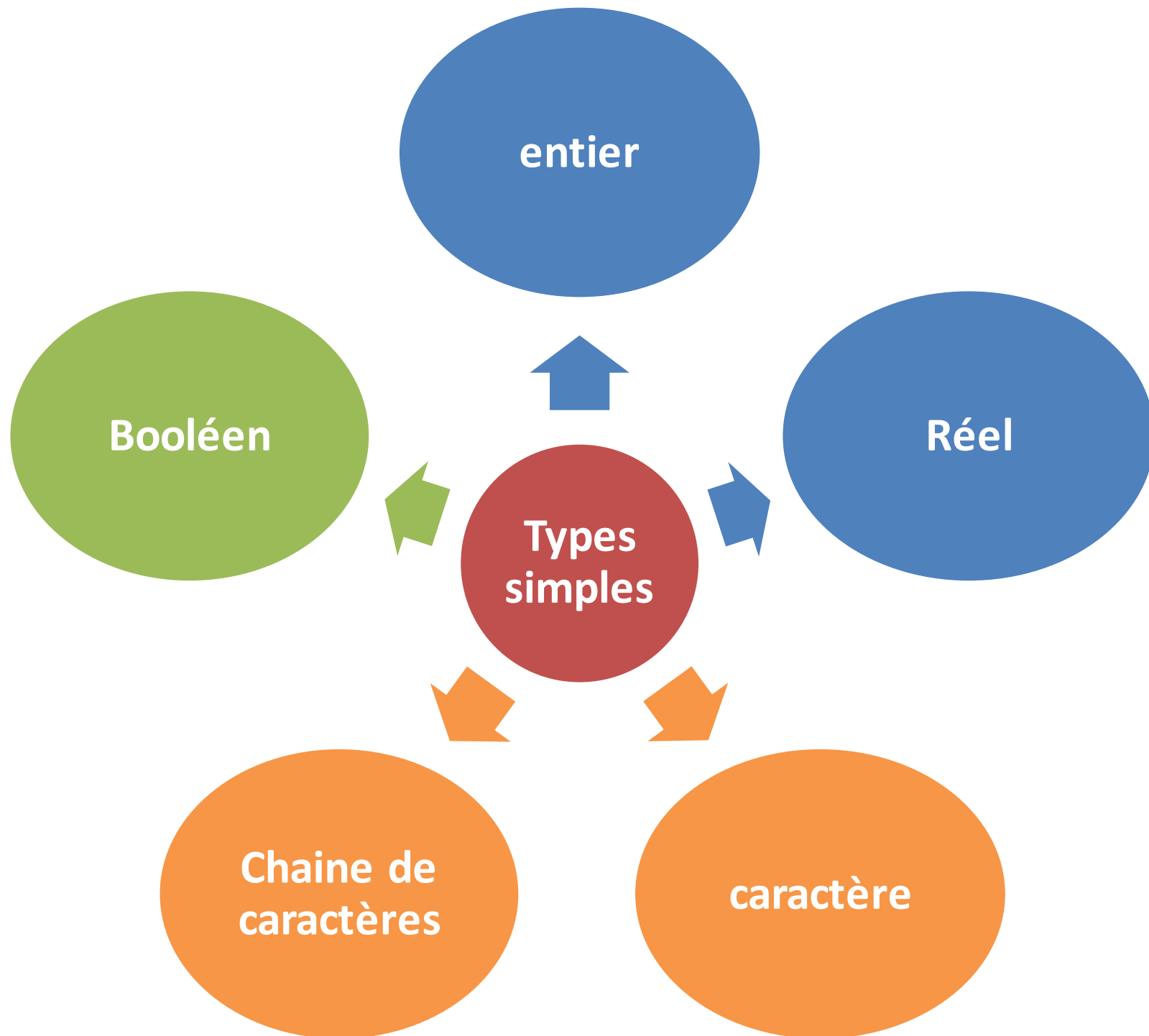
```
graph TD; A[Un type] --> B[Valeurs possibles]; A --> C[Opérations possibles]; B --> D[a b c 3,14 ...]; C --> E["+ - / * ..."]
```

Valeurs possibles

a b c 3,14 ...

Opérations possibles

+ - / * ...



Type entier

Valeurs

- entières
 - 0, 1, 2, ...
 - -1, -2, -3, ...

Entiers

Opérations primitives

- $-x$ (opposé)
- $x + y$ (addition)
- $x - y$ (soustraction)

Entiers

Opérations primitives

- $x * y$ (multiplication)
- x / y (division résultat réel)

Integer

Primitives Pascal

- $x \text{ div } y$ (division, résultat entier)
- $x \text{ mod } y$ (reste de la division)

Integer

Primitives Pascal

- **abs** (x) (valeur absolue)
- **pred** (x) (précédent de x)
- **succ** (x) (successeur de x)

Integer

Primitives Pascal

- **odd** (x) (vérifie si x est paire)
- **sqr** (x) (carré de x)
- Opérations primitives sur réels

Type réel

Valeurs

- Réelles
- Deux notations
 - 12.5 (séparateur .)
 - 125e-1 (125×10^{-1})

Réels

Opérations primitives

- $-x$ (opposé)
- $x + y$ (addition)
- $x - y$ (soustraction)

Réels

Opérations primitives

- $x * y$ (multiplication)
- x / y (division résultat réel)

Entiers & Réels

Opérations primitives

- Comparaison (résultat vrai ou faux)
 - $<$, \leq , $>$, \geq , $=$
- $<>$ (différent)

Real

Primitives Pascal

- **sqr** (x) (carré de x)
- **sqrt** (x) (racine carré de x)

Real

Primitives Pascal

- **abs** (x) (valeur absolue)
- **trunc** (x) (partie entière)
- **round** (x) (plus proche entier)

Real

Primitives Pascal

- **sin (x), cos (x), arctan (x)**
- **exp (x), ln (x)**

Utilisation de primitives?

Proposer un algorithme qui calcule le nombre de tickets de petit-déjeuner et la monnaie à partir d'un montant.

Utilisation de primitives?

Proposer un algorithme qui calcule la moyenne de deux nombres entiers.

Utilisation de primitives?

Proposer un algorithme qui calcule la valeur absolue d'un nombre entier.

Utilisation de primitives?

Proposer un algorithme qui calcule la partie entière d'un nombre réel.

Utilisation de primitives?

Proposer un algorithme qui calcule l'arrondi d'un nombre réel.

Utilisation de primitives?

Proposer un algorithme qui calcule le carré et la racine carré d'un nombre entier.

Type caractère

Valeurs

- **Un caractère entre apostrophes**
 - 'a' caractère a
 - '1' caractère 1
 - "" caractère ' "

Caractère

'A'

En Pascal

A chaque caractère, son entier



Code Ascii

65

Caractère	Code Ascii
A	65
B	66
a	97
0	48
+	43
@	63

Source: <http://www.table-ascii.com/>

Caractères

Opérations primitives

- Comparaison basée sur code ascii
 - $<$, $<=$, $>$, $>=$
- $<>$ (différent)

Char

Primitives Pascal

- **ord** (x) (code ascii de x)
- **chr** (x) (inverse de **ord**)

Char

Primitives Pascal

- **pred** (x) (précédent ascii de x)
- **succ** (x) (successeur ascii de x)

Type booléen

Valeurs

- **Vrai**
- **Faux**

Booléen

Opérations primitives

- **non** (x) (négation)
- x **et** y
- x **ou** y
- x **ouExclusif** y

Booléen

Opérations primitives

- Comparaison (résultat vrai ou faux)
 - $<$, \leq , $>$, \geq , $=$
 - $<>$ (différent)

Avec Faux $<$ Vrai

Boolean Primitives Pascal

- **not** (x) (négation)
- x **and** y
- x **or** y

x	y	<i>Not(x)</i>	<i>X and Y</i>	<i>X or Y</i>	<i>X Xor Y</i>
True	True	False	True	True	False
True	False	False	False	True	True
False	True	True	False	True	True
False	False	True	False	False	False

Table de vérité

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est strictement positif.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre n n'est pas strictement positif.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est plus grand que 100.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est strictement compris entre 1 et 100.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est strictement compris entre 1 et 100 sans être égal à 50.

Condition booléenne?

Proposer un algorithme de calcul du signe du produit de deux nombres entiers.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est plus grand que trois autres nombres.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est plus petit que deux autres nombres tout en restant plus grand que 100.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est plus petit que deux autres nombres tout en restant plus grand qu'un troisième.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est plus petit que deux autres nombres tout ou plus grand qu'un troisième.

Condition booléenne?

Proposer un algorithme qui vérifie si un nombre est compris entre deux nombres donnés.

Condition booléenne?

Proposer un algorithme qui vérifie si un caractère est une voyelle.

Condition booléenne?

Proposer un algorithme qui vérifie si un caractère est une consonne.

Type chaîne de caractères

Valeurs

- **Des caractères entre apostrophes**
 - 'bonjour'

Chaine de caractères

Opérations primitives

- **+ (concaténation)**

Chaine de caractères

Opérations primitives

- Comparaison lexicographique & ascii
 - $<$, $<=$, $>$, $>=$, $=$
 - $<>$ (différent)

String Primitives Pascal

- **length** (x) (longueur de x)
- **pos** (x,y) (position de x dans y)

String Primitives Pascal

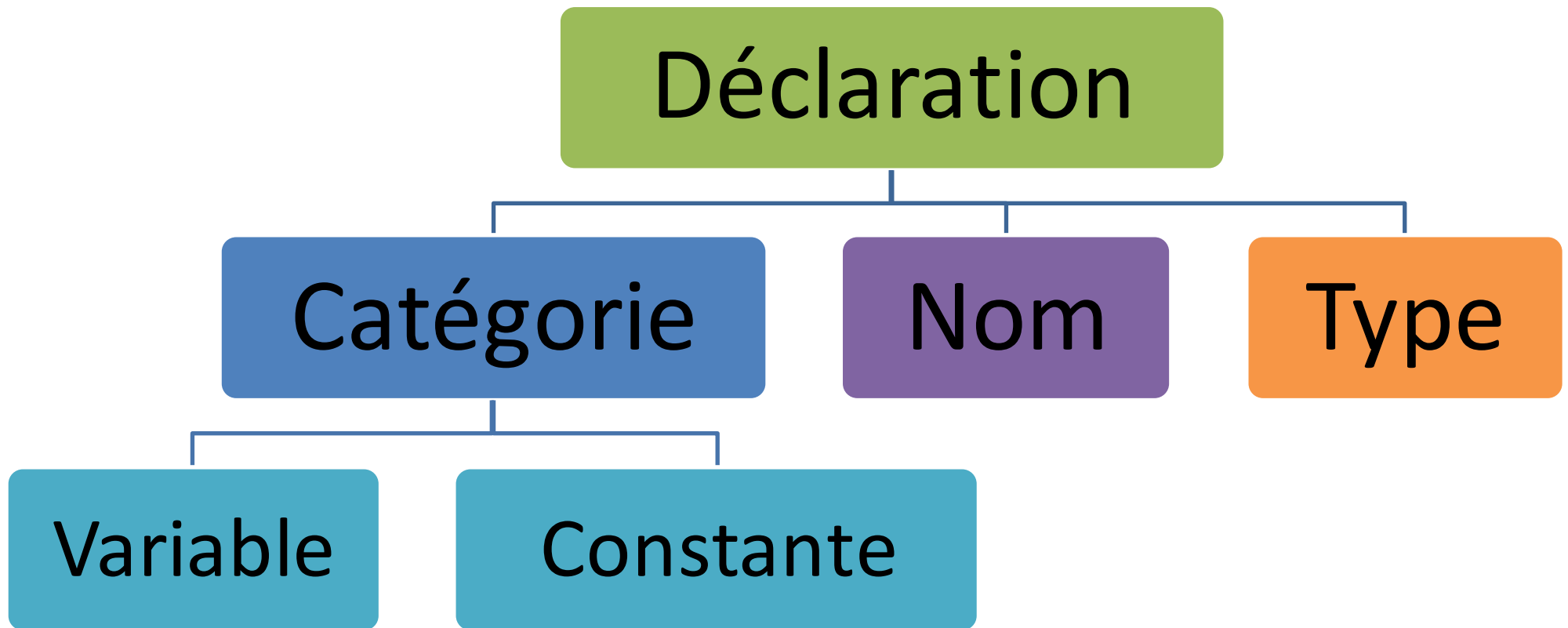
- **concat** (x,y)
- **copy** (x,debut, longueur)
- **insert** (x,dans, a partir)

String Primitives Pascal

- **x[position]** (caractère à position)

Déclaration des données

**La déclaration des données
d'entrée, de sortie et intermédiaires
est obligatoire.**



Déclaration d'une constante

Algorithme

	Constante réelle $\pi = 3,14$

Déclaration d'une constante

Pseudo code

Constante réelle $\text{PI} = 3,14$

Constante entière $\text{NUM} = 12$

Constante chaîne $\text{MESS} = \text{'Bonjour'}$

Déclaration d'une constante

Pascal

Const PI = 3.14 ;

NUM:integer = 12 ;

MESS = 'Bonjour' ;

PI = 3.14

NOM = 'Sow'

Une **constante** est initialisée avec sa **valeur** définitive lors de sa déclaration.

CREDIT_EC = 12

PRIX_TICK 75

Déclaration de variables

Algorithme

	Réel moyenne, prix

Déclaration de variables

Pseudo code

Réel moyenne

Entier cote, perimetre

Chaine universite

Déclaration de variables

Pascal

Var

moyenne:real;

cote, perimetre:integer;

universite: string;

perimetre est égale à *cote**4

PI vaut 3.14

cote est 10

**Une donnée est utile si elle
dispose d'une valeur**

largeur est 12

solution est égale à $-b/c$

PRIXTICK est 75

A une **variable** nous pouvons
principalement

affecter la valeur
d'une expression

lire sa valeur au
clavier

'Dakar'

15.12

14

'a'

NOM
largeur

a

moyenne mod 4
*cote * 4*
*(longueur + largeur) * 2*

Expression

Constante

Donnée

**Combinaison
des deux et des
opérations
primitives**

Affectation

Algorithme

perimetre = cote * 4

Affectation

PseudoCode

← **ou** ::=

perimetre ← *cote* * 4

Affectation

Pascal

:=

perimetre := cote * 4 ;

Le type de la valeur affectée doit être compatible avec celui de la donnée.

Lecture au clavier

Algorithme



lire largeur

Lecture au clavier

PseudoCode

lire(donnée)

Exemple:

lire (largeur)

Lecture au clavier

Pascal

read(donnée); (retour à la ligne, espace, tab)

readln(donnée);

readln;

read(largeur);

readln;

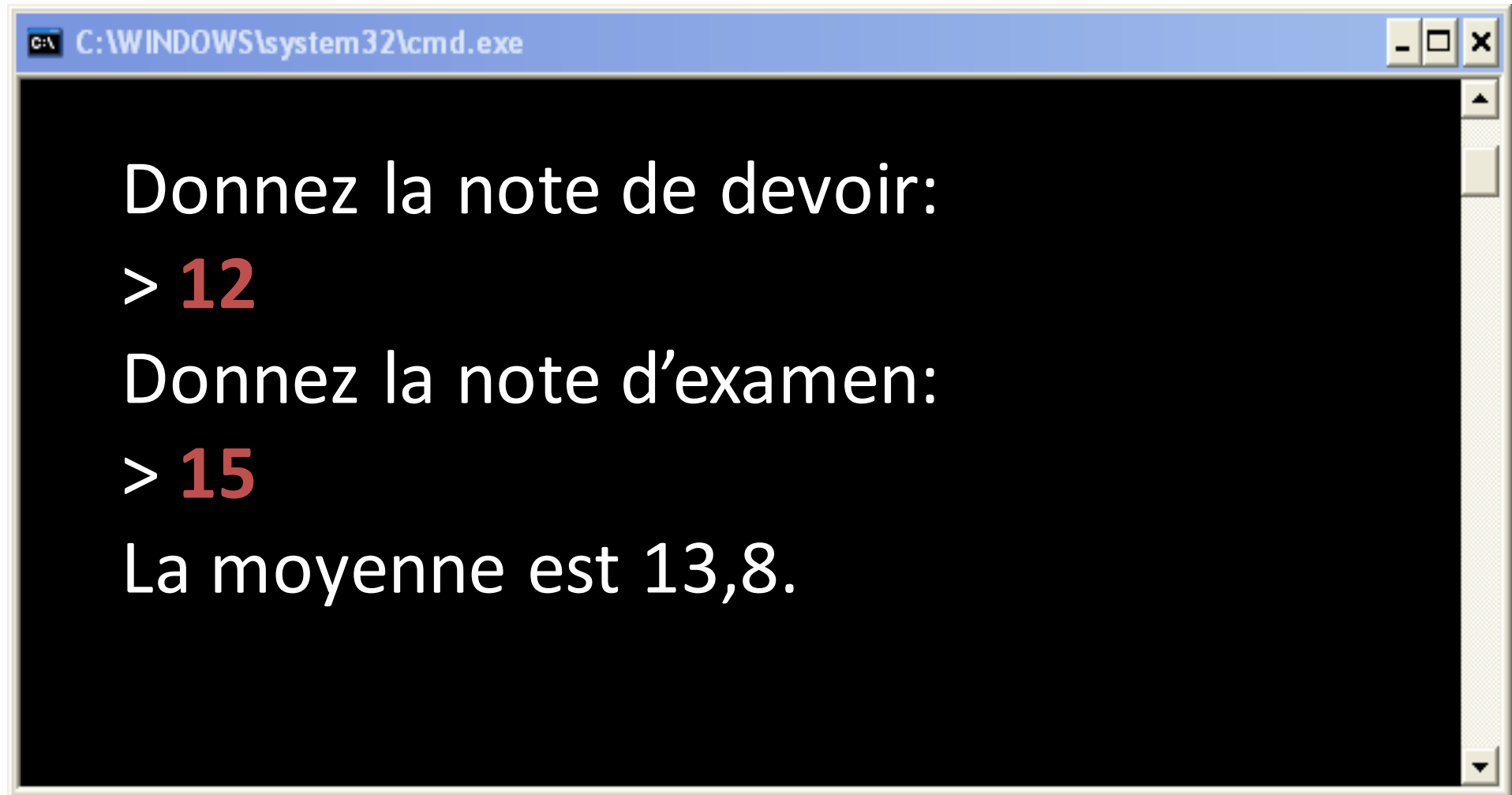
read(longueur);

readln (longueur);

read(longueur,largeur);

Le type de la valeur lue doit être compatible avec celui de la donnée.

Avec la lecture, l'utilisateur peut transmettre les données nécessaires à l'exécution de l'algorithme.



```
C:\WINDOWS\system32\cmd.exe
```

Donnez la note de devoir:

> 12

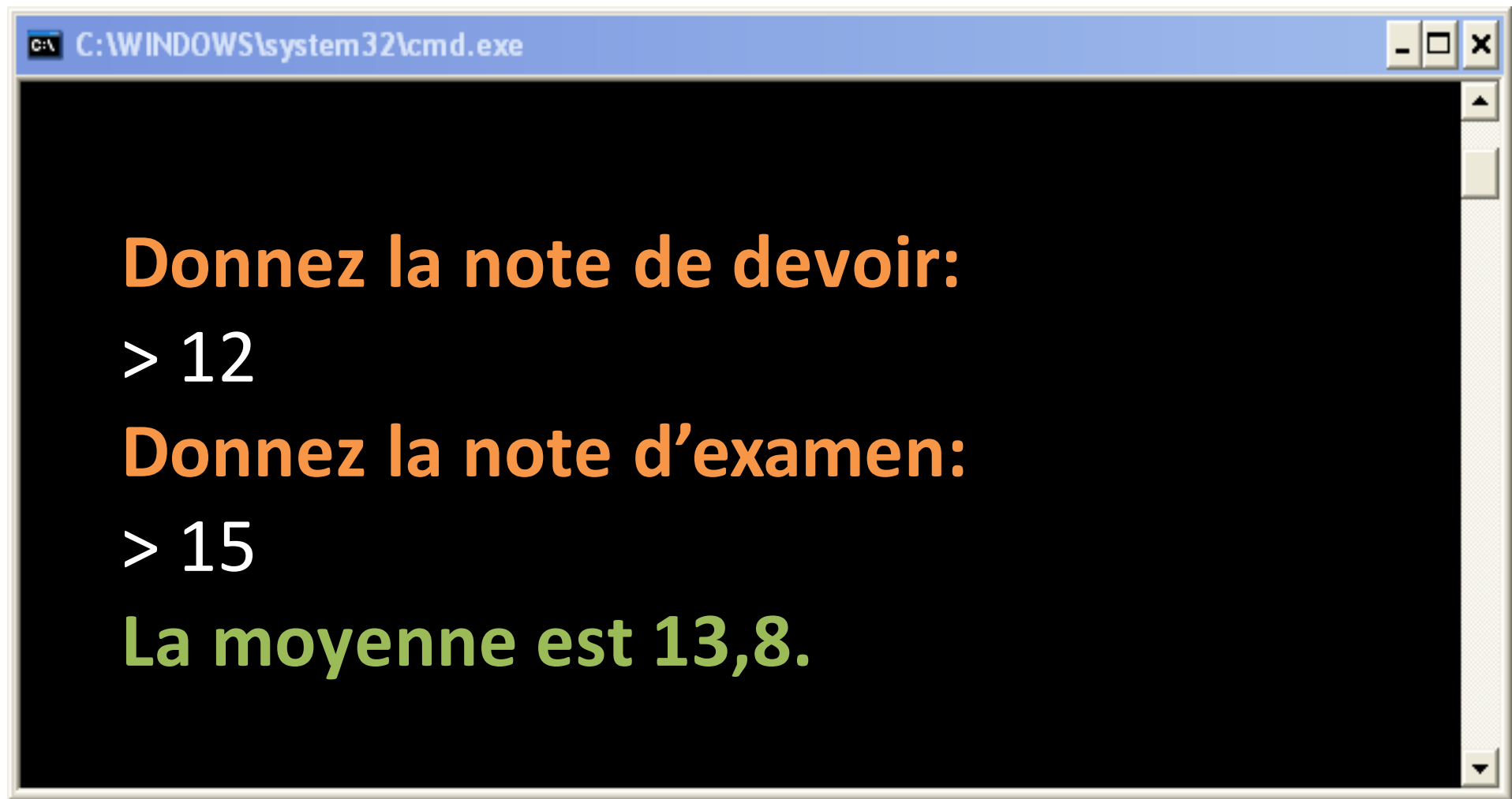
Donnez la note d'examen:

> 15

La moyenne est 13,8.

Données transmises par l'utilisateur

**Avec l'écriture, l'algorithme peut
informer l'utilisateur sur les
données finales ou sur les
actions à faire.**



Actions à faire

Résultat final

Ecriture à l'écran

Algorithme



ecrire moyenne

Ecriture à l'écran

PseudoCode

ecrire(donnée)

Exemple:

ecrire (moyenne)

Écriture à l'écran

Pascal

write

writeln

write(perimetre);

writeln;

writeln(surface);

write('Le perimetre est: ',perimetre);

write('Le perimetre est: ',perimetre:4,' et la surface:',surface);

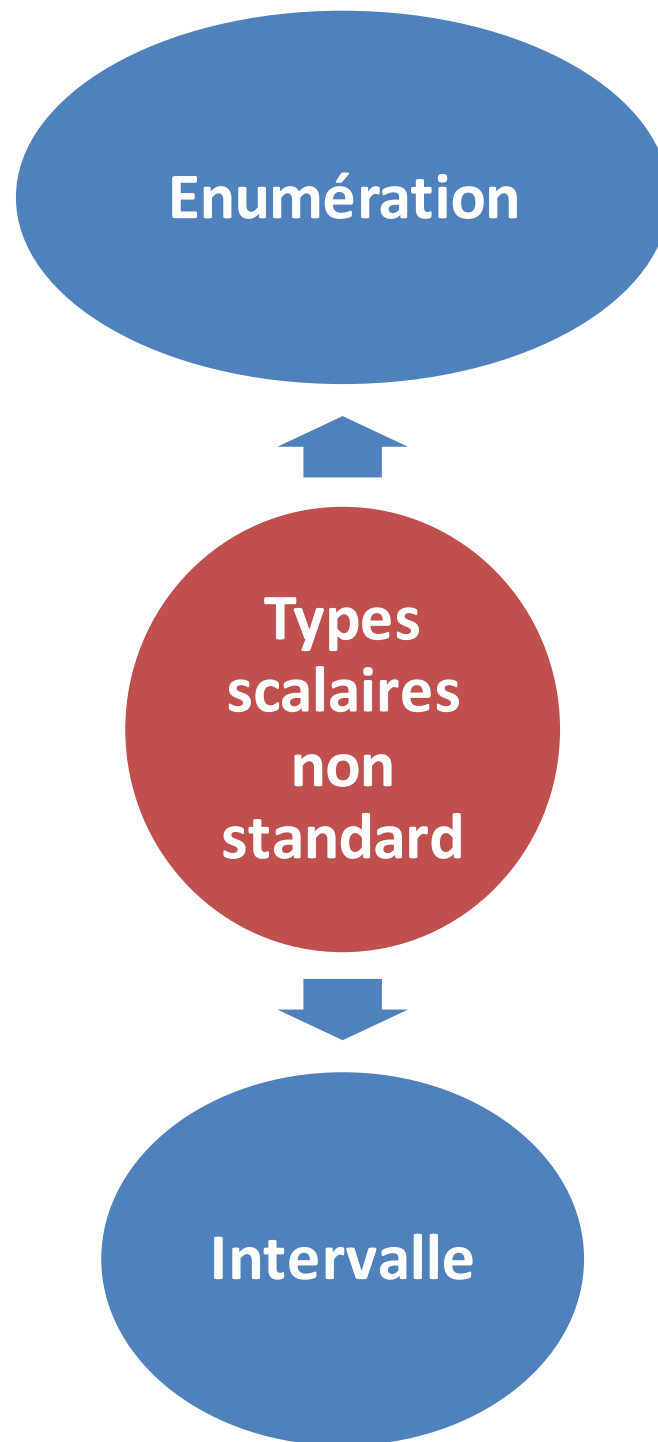
write('La moyenne est: ',moyenne:6:2);

write ou Write ou WRITE ou ...

Begin ou begin ou BEGIN ou ...

La casse n'est pas prise en
compte en Pascal **mais...**

largeur ou LarGeur ou LARGEUR ou ...



**Une énumération est une suite
d'identificateurs dont les
valeurs sont ordonnées.**

Variables énumération

Var

0 1 2

couleur1, couleur2 : (vert, jaune, rouge);

jourJ : (lundi, mardi, mercredi, jeudi);

0 1 2 3

Type énumération

Type

DrapeauSen = (vert, jaune, rouge);

Jour = (lundi, mardi, mercredi, jeudi);

Var

couleur1, couleur2 : DrapeauSen;

jourJ : Jour;

```
couleur1 := vert;  
jourJ := lundi;  
couleur2:=succ(couleur1);  
write ('Ordre couleur:', ord(couleur2));
```

Une variable énumération ne peut être ni lue ni écrite.

Un intervalle est une partie d'un type de base (entier, booléen, caractère, énuméré) composée de valeurs consécutives.

Variable intervalle

Var

moisCourant : 1 .. 12;

initialNom : 'A' .. 'R';

debSemaine : lundi .. jeudi;

chiffreCh = '0' .. '9';

Type intervalle

Type

Mois = 1 .. 12;

DebutCh = 'A' .. 'G';

Jour = lundi .. jeudi;

ChiffreChar = '0' .. '9';

Var

moisCourant : Mois;

initialNom : DebutCh;

debSemaine : Jour;

chiffreCh = chiffreChar;

```
readln(mois);  
initialNom := 'B';  
debutSemaine:=mardi;  
write ('mois:', mois, 'et initial',initialNom);  
write ('num jour', ord(debutSemaine));
```

Squelette d'un programme Pascal

```
graph TD; ENTETE[ENTETE] --- DECLARATIONS[DECLARATIONS]; DECLARATIONS --- CORPS[CORPS];
```

ENTETE

DECLARATIONS

CORPS

```
program moyenne;  
  uses crt;  
var  
  devoir, exam, moy: real;  
begin  
  writeln('Donnez le devoir');  
  readln(devoir);  
  writeln('Donnez l'examen');  
  readln(exam);  
  moy:=devoir*0.4+ exam*0.6;  
  writeln('La moyenne est', moy);  
end.
```

ENTETE

- **Program nomProg;**

DECLARATIONS

- **uses ... ;**
- **const ...;**
- **var ...;**

CORPS

- **begin**
- ...
- **end.**

Quelles sont les informations indispensables sur une donnée?

Quelles sont les catégories possibles pour une données?

**Sur quelles critères se baser
pour choisir une constante ou
une variable?**

**Quelles sont les conditions de
nommage des données?**

Citez une convention de nommage des constantes, variables simples et composées?

**Quelles est l'importance de
disposer d'une convention?**

Quelles sont les types simples disponibles en pascal?

**Donnez deux exemples de
valeurs pour chaque type
simple?**

**A quoi servent les primitives:
div, mod, abs, trunc, round, /,
trunc, sqr, sqrt, odd, length?**

Déclarez deux constantes de chaque type.

Déclarez deux variables de chaque type et affectez à l'une une valeur et saisissez l'autre.

**La casse, est elle importante
en Pascal?**

Ecrire en Pascal un programme simple de calcul de la somme de trois entiers saisis au clavier.

La somme de 1, 3, 5 est 9

Ecrire en Pascal un programme simple de calcul de la division entière et du reste de la division de deux entiers saisis au clavier.

10/3=3 et il reste 1

**Quelle est la syntaxe générale
de déclaration d'un type?**

**Déclarez le type « entier »
correspondant à integer, puis
déclarez une variable à partir
de ce type.**

**Déclarez un type « meuble »
comportant les valeurs « chaise »,
« armoire », « fauteuil », « table »
et déclarez une variable de ce
type.**

Est il possible de saisir cette variable?

Est il possible de lui affecter les valeurs 1,
fauteuil, sofa?

**Déclarez un type « note »
comportant les valeurs entières
comprises entre 0 et 20, puis
déclarez une variable de ce type.**

Est il possible de saisir cette variable?

Est il possible de lui affecter les valeurs
1,22,-5?

Déclarez un type « jour » comportant les jours de la semaine et un type jourTravail du lundi au vendredi, puis déclarez une variable de chaque type.

Est il possible de saisir ces variables?

Est il possible de leur affecter des valeurs?

Si oui, lesquels?

