

FICHE DE RÉVISION : LES TYPES NUMÉRIQUES

Informatique — 2ème Année Secondaire (Entiers & Réels)

Algorithmique Langage Python Programme Officiel Tunsien

1. Le Type Entier (Integer)

Le type **Entier** est utilisé pour manipuler des nombres sans virgule (positifs, négatifs ou nuls).

En Algorithmme

- Type : `Entier`
- Exemples : `15`, `-8`, `0`, `2026`

En Python

- Type : `int`
- Exemples : `x = 15`, `y = -8`

Opérateurs Arithmétiques sur les Entiers

OPÉRATION	ALGORITHME	PYTHON	EXEMPLE	RÉSULTAT
Addition / Soustraction	<code>+</code> / <code>-</code>	<code>+</code> / <code>-</code>	<code>12 + 5</code>	<code>17</code>
Multiplication	<code>*</code>	<code>*</code>	<code>4 * 3</code>	<code>12</code>
Division Entière	<code>DIV</code>	<code>//</code>	<code>17 DIV 5</code> / <code>17 // 5</code>	<code>3</code>
Modulo (Reste)	<code>MOD</code>	<code>%</code>	<code>17 MOD 5</code> / <code>17 % 5</code>	<code>2</code>

💡 Usages incontournables de DIV et MOD en examen

- Parité d'un nombre N** : N est pair si `N MOD 2 = 0` (en Python : `N % 2 == 0`).
- Divisibilité de A par B** : A est divisible par B si `A MOD B = 0`.
- Isoler les chiffres d'un nombre à 2 digits (ex: N = 47)** :
 - Chiffre des unités : `U = N MOD 10` (donne 7)
 - Chiffre des dizaines : `D = N DIV 10` (donne 4)

2. Le Type Réel (Float)

Le type **Réel** représente les nombres décimaux (à virgule).

En Algorithmme

- Type : `Réel`
- Exemples : `3.14`, `-0.5`, `12.0`

En Python

- Type : `float`
- Séparateur : le point `.` (jamais la virgule)

Opérateurs Arithmétiques sur les Réels

OPÉRATION	ALGORITHME	PYTHON	EXEMPLE	TYPE DU RÉSULTAT
Addition / Soustraction	<code>+</code> / <code>-</code>	<code>+</code> / <code>-</code>	<code>5.5 + 2.3</code> → <code>7.8</code>	Réel (<code>float</code>)
Multiplication	<code>*</code>	<code>*</code>	<code>2.5 * 4</code> → <code>10.0</code>	Réel (<code>float</code>)
Division Réelle	<code>/</code>	<code>/</code>	<code>7 / 2</code> → <code>3.5</code>	Réel (<code>float</code>)

⚠ Règle d'or sur la division /

En Python comme en algorithmique, le résultat d'une division simple `/` est **TOUJOURS un Réel (float)**, même si la division tombe juste ! Exemple : `8 / 2` donne `4.0` et non `4`.

3. Fonctions Standards Usuelles

Ces fonctions prédéfinies permettent d'effectuer des traitements mathématiques sur les entiers et les réels.

RÔLE / ACTION	ALGORITHMME	PYTHON	EXEMPLE & RÉSULTAT
Valeur Absolue	<code>ABS(x)</code>	<code>abs(x)</code>	<code>ABS(-9)</code> → 9
Racine Carrée	<code>RACINE_CARRE(x)</code> ou <code>SQRT(x)</code>	<code>math.sqrt(x)</code> *	<code>SQRT(16)</code> → 4.0
Partie Entière (Troncature)	<code>ENTIER(x)</code> ou <code>TRUNC(x)</code>	<code>int(x)</code> ou <code>math.trunc(x)</code>	<code>ENTIER(7.9)</code> → 7
Arrondi au plus proche	<code>ARRONDI(x)</code>	<code>round(x)</code>	<code>round(3.6)</code> → 4 <code>round(3.2)</code> → 3
Puissance x^y	<code>POUISSANCE(x, y)</code>	<code>x ** y</code> ou <code>pow(x, y)</code>	<code>2 ** 3</code> → 8
Nombre Aléatoire	<code>ALEA(a, b)</code>	<code>random.randint(a, b)</code> **	<code>ALEA(1, 6)</code> → entier entre 1 et 6

* Nécessite l'instruction `import math` en début de programme Python.

** Nécessite l'instruction `import random` en début de programme Python.

4. Conversions de Type & Erreurs à Éviter

Conversion de Type (Casting)

Pour convertir une saisie utilisateur (qui est par défaut une chaîne de caractères) en nombre :

- En Entier :
`n = int(input("Donner N: "))`
- En Réel : `x = float(input("Donner X: "))`
- Un réel vers un entier : `int(3.85)` → 3

Pièges Classiques d'Examen

- **Division par Zéro** : `X / 0` ou `X MOD 0` provoque une erreur fatale (`ZeroDivisionError`).
- **MOD/DIV sur Réels** : Interdit en algorithmique classique ! `5.5 MOD 2` est incorrect.
- **Racine d'un négatif** : `SQRT(-4)` déclenche une erreur d'exécution (`ValueError`).

5. Exercices d'Application Express

Exercice 1 : Évaluer les expressions suivantes

Donner le résultat et le type des expressions ci-dessous :

1. `A = 19 DIV 4` `B = 19 MOD 4`
2. `C = 19 / 4` `D = ABS(-5) * (2 ** 3)`

Correction :

1. `A = 4` (Entier) | `B = 3` (Entier)
2. `C = 4.75` (Réal) | `D = 5 * 8 = 40` (Entier)

Exercice 2 : Traduction en Python

Traduire la formule mathématique suivante en Python : $R = \sqrt{a^2 + b^2}$

Solution Python :

```
import math  
R = math.sqrt(a**2 + b**2) ou R = (a**2 + b**2) ** 0.5
```