

Découverte et prise en main



Exercice 1 : Caractérisation du PC utilisé

Partie 1 : Principaux composants d'une machine numérique

Un PC ouvert est mis à votre disposition. L'ordinateur est un matériel communiquant.

☞ Observer les éléments du panneau arrière.

Désignation		Fonction	Entrée/ sortie	Vitesse	Appeler à disparaître
Port PS/2		Communication clavier souris	S		Supplanté par USB ou bluetooth
Port Série COM1		Communication série	E/S	19200 bps	Supplanté par USB
Port Parallèle		Communication parallèle (imprimante)	S	16 Mbps	Supplanté par USB
Port USB		Communication série	E/S	USB2 60 Mo/s	
Port VGA de la carte vidéo intégré		Communication vidéo analogique	S		
Port DVI		Communication vidéo numérique non HD	S		
Port HDMI		Communication vidéo numérique HD	S		
Port Réseau RJ45		Connexion filaire au réseau ethernet	E/S	De 128 Mo/s à 12800 Mo/s	
Connecteurs sons		Connexion aux hauts parleurs externes et micro	E/S		

Q1 : Identifier les ports de communication présents et absents.

Partie 2 : Caractéristiques principales de la machine numérique utilisée

☞ Allumer l'ordinateur à votre disposition.

☞ Utiliser vos identifiants pour pouvoir ouvrir une session et utiliser le PC.

Q2 : Rechercher le nom et la version du système d'exploitation de la machine.

Q3 : Chercher si le processeur de cette machine est un processeur 32 bits ou 64 bits.

Q4 : L'ordinateur dispose-t-il de mémoire(s) de masse ? Quelle est la nature et la taille de celle(s)-ci ?

Q5 : Quelle est la taille de la mémoire vive ?

Partie 3 : Organisation des fichiers

☞ Cliquer sur Démarrer, puis Mes Documents.

☞ Ce répertoire est votre espace de stockage personnel accessible depuis chaque poste.

Q6 : Ce répertoire est-il physiquement situé sur le poste de travail ou ailleurs ?

☞ Créer un répertoire Mes Fichiers Python. (clic droit, puis nouveau / dossier)

☞ Créer un fichier texte bidon dans ce répertoire. (clic droit, puis nouveau, document texte)

☞ Aller sur la fenêtre Ordinateur.

☞ Consulter les dossiers du disque dur U :

☞ Essayer d'accéder à un répertoire. Normalement vous n'avez les droits d'accès que pour votre compte. Y a-t-il le fichier bidon ?

☞ Retrouver votre fichier bidon dans l'arborescence de U :

Q7 : Les répertoires « Mes Documents » et « U : » sont-ils identiques ou différents ?

Partie 4 : Environnement dos

☞ Cliquer démarrer/tous les programmes/accessoires/Invite de commandes
 Une fenêtre s’ouvre. C’est une interface qui utilise uniquement le clavier.

Si « U:\ » s’affiche : cela signifie que « vous vous trouvez » à la racine du disque « U : ».
 Si « U:\toto\tata\ » s’affiche : cela signifie que « vous vous trouvez » dans le répertoire « tata » du répertoire « toto » du disque « U : ».

Il est alors possible de lister les répertoires et de naviguer en utilisant les commandes :

- dir : vous permet d’avoir la liste des fichiers et répertoires où vous vous trouvez
- cd Nomdurépertoire : vous permet de changer de répertoire
- cd .. : vous permet de revenir au répertoire contenant ou amont
- x : : vous permet de changer de disque vers le disque « x ».
- del nomdufichier : vous permet de détruire.

☞ Avec ces trois commandes, retrouver votre fichier « bidon.txt » et détruisez-le.

Exercice 2 : Présentation de la console

Partie 1 : Prise en main

Double cliquer sur le raccourci « Idle » situé sur le bureau.
 La fenêtre qui s’ouvre est appelé « console ».

Q1 : A l’ouverture, la console indique la version de Python utilisée. Relever la version de Python utilisée en précisant 32 bits ou 64 bits.

Nous allons utiliser la console interactive de Python où chaque ligne tapée est immédiatement exécutée. L’interpréteur s’utilise comme une calculatrice : Il suffit de taper une instruction, pour qu’elle s’exécute. S’il y a une réponse, elle sera affichée.

Il est aussi possible d’invoquer l’aide d’une fonction en tapant : `help (nom de la fonction)`

Partie 2 : Découverte et distinction des types

Les valeurs en Python sont *typées*, autrement dit, elles sont classées selon l’objet qu’elles représentent. Une valeur peut ainsi être de type entier, flottant, booléen, chaîne de caractères, etc.

Pour connaître le type d’une valeur, il est possible d’utiliser la commande (ou fonction) :

<code>type(object) -> string</code>	Renvoie le type de l’objet
----------------------------------------	----------------------------

Exemple : taper :

```
>>> type(2)
```

Où l’on constate que le type de la valeur de a est le type entier (*integer*, en anglais).

Q2 : En utilisant la même procédure, déterminer le type des valeurs suivantes:

valeurs	type
False	
486	
True	
-32	
-32.	
3.14	
"PCSI rules"	

Partie 3 : Manipulation d'entiers relatifs

Exécuter les commandes suivantes :

```
>>> 3 + 5
>>> 6 - 8
>>> 2 * 3
>>> 2 ** 3
```

3.A. Listes des opérations

+	addition
-	soustraction
*	produit
x / y	division (le résultat est un flottant)
x // y	quotient de la division euclidienne de x par y
x % y	reste de la division euclidienne
-x	opposé
abs(x)	valeur absolue
x ** y	puissance

3.B. Exemple

```
>>> 125 * 2 + 5 % 3
>>> 5**2
```

3.C. Application 1

Un angle est mesuré avec un rapporteur électronique avec une précision de 1°. Cette mesure est alors codée par un entier.

Lors d'un déplacement, la mesure indique 5282°.

Q3 : Déterminer le nombre entiers de tours parcourus à la fin des déplacements.

Q4 : Déterminer l'angle final modulo 360°

3.D. Application 2

Q5 : Justifier le résultat de l'instruction : $2^{1/2}$

Partie 4 : Manipulation des nombres à virgules flottantes

Les opérations sur les nombres à virgules sont identiques à celle sur les entiers.

Taper :

```
>>> type(42 + 3.1415)
```

Q6 : Justifier l'appellation « typage dynamique »

4.A. Quelques surprises à justifier

Q7 : Comparer les résultats des opérations ; Commenter :

```
>>> 2 ** 100
>>> 2.0 ** 100
```

Q8 : Effectuer instructions suivantes et commenter les résultats sans les justifier

```
>>> 0.1+0.1-0.2
>>> 0.1+0.1+0.1-0.3
```

4.B. Application

Q9 : Déterminer le rayon d'un cylindre de 1L et de hauteur 10 cm.

4.C. Manipulation

Soient $a = 25$ et $b = 3$.

- Calculer $c = a/b$ par la division classique,
- $d = a/b$ par la division entière et le reste r de cette division.
- Calculer a^2 , \sqrt{a}

Partie 5 : Formatage

En informatique, quelques convention d'écriture sont à connaître. Prendre connaissance de la fiche associée au formatage des nombres.

5.A. Découverte

Quel est le type et la valeur décimales des saisies suivantes sous python :

Saisie	Type	Valeur décimale
1000		
0b1000		
0x1000		
1000.		

5.B. Conversion

Quelle est la valeur décimale de $(110011)_2$?

Quelle est la valeur décimale de $(110011)_{16}$?

5.C. Formatage d'un résultat

Q10 : Comparer les résultats des opérations :

```
>>> format(455**45, '.100f')
>>> format(455**45, '.100e')
```

Q11 : Afficher la racine carrée d'un milliard en notation scientifique avec 3 chiffres après la virgule

5.D. Synthèse

Q12 : A l'aide de python, déterminer le résultat de la somme $(11101010)_2 + (2b45c)_{16}$. Exprimer cette valeur en décimale, en binaire, puis en hexadécimale.

Partie 6 : Manipulations des booléens

Un booléen est le codage d'une information sur un bit : vrai (True) ou faux (False).

Il peut être obtenu par des opérations de comparaisons entre deux valeurs.

Exemple : taper :

```
>>> 2>3
```

Opération	Signification
<	Strictement inférieur
<=	Inférieur ou égal
>	Strictement supérieur
>=	Supérieur ou égal
==	Egal
!=	Différent
is	Objet égal

Q13 : Comparer et justifier le résultat des comparaisons :

```
1 == 1.0                      1 is 1.0
```

Q14 : Commenter le résultat de l'instruction :

```
>>> 0.1+0.1+0.1 == 0.3
```

Les opérations de bases sur les booléens sont :

Opération	Résultat
x or y	est vrai si x ou y est vrai
x and y	est vrai si x et y sont vrais
not x	est vrai si x est faux, et inversement

Q15 : En utilisant une commande sous Python, déterminer si $678/123$ est plus petit que $6789/1234$.

Partie 7 : Codage des caractères

Le codage des caractères est défini par le tableau ASCII. Chaque caractère est codé par un entier sur 7 bits. Il est possible de retrouver le caractère associé avec l'entier en utilisant l'instruction `chr` ou `format`, et inversement avec l'instruction `ord`.

```
>>> chr(97)
>>> format(97, 'c')
>>> chr(33)
>>> chr(128)
>>> ord(a)
>>> ord('a')
```

Q16 : Analyser les résultats des expressions précédente, et commenter.

☞ Maintenir enfoncer Alt, taper 97 (avec le pavé numérique), relâcher Alt
 Il est donc possible d'utiliser le code ASCII directement à partir d'un clavier.

Un mot, une phrase sont définis comme une chaîne de caractère, qui est une liste d'entier.

Q17 : Quel est le résultat des instructions ci-dessous ?

```
>>> a = "yop"
>>> len(a)
>>> a.upper()
```

7.A. Concaténation de chaînes

En Python, il est possible de concaténer plusieurs chaînes de caractères afin d'en créer une nouvelle. Pour cela, on utilise l'opérateur `+`.

Q18 : Quel est le résultat des instructions ci-dessous ? Commenter

```
>>> "Concaténer " + "deux chaînes"
>>> 10+2
>>> '10'+ '2'
>>> "10"+"2"
>>> "10" * 3
```

Q19 : Quelle est la chaîne de caractère associée à la liste d'entiers : 80 67 83 73 33.

Synthèse

Conversion en entier relatif	<code>int(x)</code>
Conversion en entier relatif long	<code>long(x)</code>
Conversion en nombre à virgule flottante	<code>float(x)</code>
Conversion en chaîne de caractère	<code>str(x)</code> : transformer un nombre en chaîne de caractère
	<code>format(x, 'option')</code> : transformer un nombre en chaîne de caractère en respectant un format bien précis