

TP n°5 : Introduction aux bibliothèques

Instruction générale : hormis pour les exercices corrigés collectivement, vous ferez valider votre travail par l'enseignant.

1 Bibliothèques

Les bibliothèques (ou modules) constituent des ensembles de fonctions vérifiées et validées. Elles sont mises à disposition afin de pouvoir être utilisées sans avoir à les réécrire. On appelle les bibliothèques dans un programme à l'aide de la commande `import`. Du point de vue de la structure du code, il est préférable d'importer tous les bibliothèques dont on a besoin dès le début du programme. Il existe plusieurs façon d'effectuer un `import` comme on peut le voir dans l'exemple ci-dessous avec la bibliothèque `math` qui contient des fonctions mathématiques. On peut retrouver la documentation la librairie `math` [ici](#).

- `import math`

Le module `math` est importé. On peut utiliser ses fonctions en les préfixant avec `math`. Par exemple, `sqrt()` qui désigne la racine carrée (square root en anglais) :

```
x = math.sqrt(64)
```

- `from math import *`

Toutes les fonctions du module `math` sont importées. On peut les utiliser sans préfixe.

```
x = sqrt(64)
```

- `from math import sqrt`

On importe juste ce dont on a besoin, ici la fonction `sqrt`.

```
x = sqrt(64)
```

Exercice 1. Dans la bibliothèque `math`, importer la fonction permettant de calculer la partie entière d'un nombre puis calculer celle de 7.34891 et -4.9736 .

Exercice 2. Dans la bibliothèque `math`, importer la constante π puis l'utiliser pour écrire une fonction calculant la surface et le volume d'une sphère de rayon donné.

2 Librairie random

On trouvera la documentation la librairie `random` [ici](#). Celle-ci est dédiée à la génération de nombres pseudo-aléatoires. En effet, l'aléatoire n'existe pas réellement en informatique et on doit faire appel soit à des suites de nombres construits par algorithmes de façon à « imiter » l'aléatoire (d'où pseudo-aléatoire), soit faire appel à des suites de nombres construits à partir de phénomènes physiques ayant des caractéristiques aléatoires.

Exercice 3. Dans l'interpréteur Python, importer la bibliothèque `random` puis exécuter les commandes ci-dessous plusieurs fois chacune. Que constate-t-on ?

1. `randint(1,10)`;
2. `random()`;
3. `uniform(1,10)`.

Exercice 4.

1. Écrire une fonction simulant le lancé d'un dé à six faces.
2. Écrire une fonction simulant le lancé d'un dé à un nombre quelconque de faces.

Exercice 5. [Surprise] Vous venez d'acheter une boîte de dragées surprises de Bertie Crochue. La boîte mentionne qu'elle contient cinq saveurs de dragées avec leurs proportions ; elles sont répertoriées dans le tableau suivant :

Saveurs	Brocoli	Cerise	Chocolat	Poubelle	Savon
Proportion	20%	30%	30%	5%	15%

Écrire une fonction `surprise()` simulant la prise au hasard d'une dragée dans la boîte.

Exercice 6. [Jeu de hasard] En mangeant négligemment vos dragées goût poubelle, vous êtes entré dans le casino possédé par la mafia des belettes. Or la consommation de dragées surprises est strictement prohibée dans le casino. Les belettes vous capture et vous force à programmer leurs nouvelles machines à sous. Elles veulent que les gains (mises comprises de deux sesterces) et leurs probabilités se fassent selon le tableau suivant :

Gain	-2	0	2	10	100
Probabilité	80%	10%	5%	4%	1%

Programmez une fonction `banditManchot()` satisfaisant les belettes puis sortez du casino.

Exercice 7. [Encore un jeu] Écrire un jeu auquel l'utilisateur jouera selon les règles suivantes :

- l'ordinateur prend au hasard un nombre entier entre un et cent que l'utilisateur doit deviner ;
- l'utilisateur propose une réponse à l'ordinateur une réponse, tant que ce n'est pas la bonne, il doit continuer d'en proposer ;
- si la réponse de l'utilisateur est plus grande, resp. plus petite, que le nombre choisi par l'ordinateur, ce dernier doit indiquer « C'est plus petit. », resp. « C'est plus grand. », avant de prendre la prochaine proposition.
- si la réponse est bonne, l'ordinateur doit répondre « Bravo, vous avez trouvé! ».

Exercice 8. [Des jeux, toujours des jeux] Écrire un programme jouant à votre place au jeu précédent. Il devra donc trouver le plus rapidement possible un nombre pris au hasard entre 1 et 100 tout en sachant pour chaque proposition si celle-ci est supérieure, inférieure ou égale au nombre à deviner.

3 Librairie matplotlib

La bibliothèque matplotlib n'est pas une bibliothèque standard Python mais elle est très utilisée car elle permet de tracer des graphiques. On trouvera ici sa documentation officielle.

Exercice 9.

1. Écrire le code ci-dessous dans l'interpréteur. Que fait-il ?
2. Quelles sont les coordonnées des extrémités de la courbe tracée ?

```
import matplotlib.pyplot as plt
plt.plot([0,10],[0,5])
plt.show()
```

Exercice 10.

1. Modifier le code précédent pour tracer une courbe passant les points de coordonnées $(-4; 1)$, $(-3; 2)$, $(-1; -1)$, $(2; 1)$, $(3; 1)$ et $(4; 2)$.
2. Qu'observez-vous ?

Exercice 11. Tracer une courbe de la fonction sinus pour $x \in [-10; 10]$ en plaçant des points tous les 0,01. *Indication* : on pourra créer une liste X contenant toutes les abscisses des points à afficher puis une liste Y contenant toutes les ordonnées et enfin utiliser la commande `plt.plot(X,Y)`.

Exercice 12. Tracer sur $[-10; 10]$ la courbe représentative de la fonction

$$f : x \mapsto x^3 - 2x^2 - 3x + 4.$$

On placera 1000 valeurs entre -10 et 10 . Légènder les axes et la courbe.

Exercice 13.

1. Tracer 1000 points générés aléatoirement. L'abscisse et l'ordonnée de ces points sont dans l'intervalle $[-1; 1]$ avec des valeurs au centième. *Indication* : on pourra utiliser la fonction `plt.scatter(x,y)` pour ne plus tracer une courbe mais uniquement les points.
2. Colorer de différentes couleurs :
 - les points d'abscisse et d'ordonnée négatives ;
 - les points d'abscisse négative et d'ordonnée positive ;
 - les points d'abscisse et d'ordonnée positives ;
 - les points d'abscisse positive et d'ordonnée négative.

Indication : on pourra utiliser la fonction `plt.scatter(x,y)` 4 fois.

Exercice 14. [TVA] Le tableau ci-dessus donne le poids de la TVA relativement aux revenus des ménages (« taux d'efforts ») par décile de niveau de vie en 2011 en France.

Décile	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Taux d'effort (en %)	12,5	9,6	9	8,5	8,1	8	7,8	7,7	7,2	4,7

Tracer un histogramme avec légende illustrant ces résultats. Que peut-on en déduire ?

Exercice 15. [Température] Chicago et Rome sont situées à la même latitude. Voici les relevés des températures moyennes à la mi-journée des deux villes en 2017.

Mois	J	F	M	A	M	J	J	A	S	O	N	D
Rome (en °C)	11	16	18	20	23	28	34	36	25	23	16	12
Chicago (en °F)	32	39	41	57	59	75	82	79	72	57	39	27

Le lien entre degré Celsius et Fahrenheit est donnée par la formule suivante :

$$T^{\circ}F = \left(\frac{5}{9} \times T^{\circ}C\right) + 32.$$

1. Sur le même graphique, tracer en rouge la courbe des températures de Rome et en bleu celle de Chicago.
2. Ajouter une droite bleue et une droite rouge représentant la température annuelle moyenne respectivement de Chicago et de Rome.

Exercice 16. [Dépenses publiques] Le site À quoi servent mes impôts permet de visualiser comment est reparti l'argent de nos impôts en fonction des différentes dépenses de l'État. Pour 1000€ dépensés en 2019, on a la répartition suivante :

Protection sociale : 572€ :

- Retraites : 262€;
- Santé : 192€;
- Famille : 41€;
- Chômage : 33€;
- Logement : 15€;
- Autres : 26€.

Régalien : 60€ :

- Défense : 31€;
- Justice : 4€;
- Sécurité : 25€.

Recherche : 23€.

Administration : 66€.

Charge de la dette : 28€.

Éducation : 95€.

Dépenses sectorielles : 156€ :

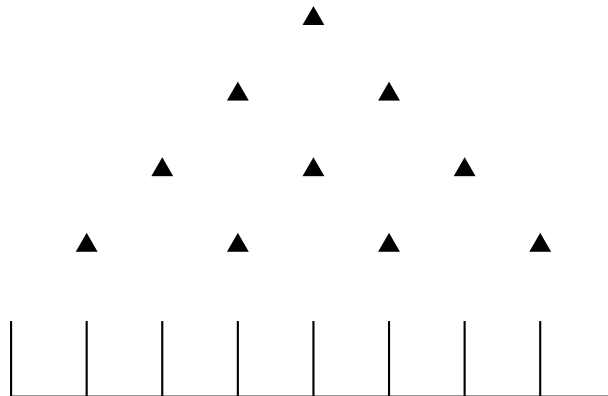
- Affaires économiques : 58€;
- Culture : 26€;
- Environnement : 17€;
- Infrastructures : 8€;
- Transports et équipements : 47€.

Calculer les parts des différents secteurs dans la dépense publique et les représenter dans un diagramme circulaire légendé.

Exercice 17. [L'impôt, la prestation et le voisin] Sur la même figure, tracer l'histogramme des pourcentages de prélèvements obligatoires (impôts, taxes, cotisations) et de prestations sociales en 2017 des pays du tableau ci-dessous. Légendé la figure.

Pays	France	Belgique	Allemagne	Royaume-Uni	Espagne
P.O. (en % de PIB)	46.1	44.5	37.6	33.3	33.7
P.S. (en % de PIB)	34.1	28.8	29.7	26.3	24.3

Exercice 18. [Planche de Galton] Une planche de Galton est une planche verticale sur laquelle sont disposés des « piquets » sous formes de pyramides comme sur le schéma ci-dessous avec une pyramide à quatre étages.



Les planches de Galton sont faites de façon à ce qu'une bille tombant sur l'un des piquets ait autant de chances de tomber à gauche qu'à droite. En tombant, elle va nécessairement sur un autre piquet jusqu'à temps qu'elle ait traversé toute la pyramide. Une fois la pyramide traversée, la bille tombe dans un des récipients situés en dessous. On peut alors compter le nombre de billes dans chaque récipient pour déterminer la répartition des billes.

1. Créer une fonction `plancheGalton()` simulant la chute d'une bille dans une planche de Galton de taille quelconque. *Indications* : faire le lien entre le nombre d'étages de la planche et le nombre de positions possibles de la bille puis raisonner en terme de position étape par étape.
2. Simuler la chute de 1000 billes dans la planche de Galton et enregistre le nombre de billes arrivant dans chacune des positions finales.
3. Afficher le résultat sous forme d'histogramme.
4. Reproduire plusieurs fois l'expérience (en faisant varier le nombre de billes ou la taille de la planche éventuellement). Qu'en déduisez-vous ?

Exercice 19. [Impôt sur le revenu] L'impôt sur le revenu se calcule à partir des revenus annuels d'une personne avec un fonctionnement par tranches. En 2021, les tranches et taux d'impositions pour celles-ci étaient donnés par le tableau ci-dessous.

Tranches	$[0; 10064[$	$[10064; 25660[$	$[25660; 73370[$	$[73370; 157807[$	$[157807; +\infty[$
Taux d'imposition	0%	11%	30%	41%	45%

Exemples :

- Pour un salaire annuel de 10000€. Le salaire est inférieur à 10064€ donc on est dans la première tranche; on est non imposable.
- Pour un salaire annuel de 20000€. L'erreur est de considérer que l'on est taxé à 11%. Le salaire est en fait découpé selon les tranches :

Tranches	$[0; 10064[$	$[10064; 25660[$
Calcul du montant dans la tranche	$10064 - 0 = 10064$	$20000 - 10064 = 9036$
Taux d'imposition	0%	11%
Montant prélevé sur la tranche	$10064 \times 0\% = 0$	$9036 \times 11\% = 993,96$

Pour un salaire annuel de 20000€, on doit donc payer 993,96€. Cela représente 4,96% du salaire annuel et non 11%.

- Un salaire annuel de 200000€ sera découpé ainsi :

$$\begin{aligned}
 200000 &= (10064 - 0) + (25660 - 10064) + (73370 - 25660) + (157807 - 73370) \\
 &\quad + (200000 - 157807) \\
 &= \underbrace{10064}_{\text{tranche 1}} + \underbrace{15596}_{\text{tranche 2}} + \underbrace{47710}_{\text{tranche 3}} + \underbrace{84437}_{\text{tranche 4}} + \underbrace{42193}_{\text{tranche 5}}.
 \end{aligned}$$

Le montant à prélever est alors calculé comme suit :

$$10064 \times 0\% + 15596 \times 11\% + 47710 \times 30\% + 84437 \times 41\% + 42193 \times 45\% = 69634,58,$$

soit un taux d'imposition réel de 34,82%.

1. Écrire un programme permettant de calculer le montant des impôts sur le revenu, le taux d'imposition réel associé et les revenus restants après imposition pour un salaire annuel quelconque, des tranches d'impositions quelconques et des taux d'impositions par tranches quelconques.
2. Sur deux graphiques différents, afficher pour des salaires annuels allant de 1000€ à 1000000€ en augmentant de 1000 en 1000 :
 - (a) le montant prélevé et le montant restant après prélèvement ;
 - (b) le taux d'imposition réel.
3. Reproduire l'expérience en changeant (à votre convenance) les tranches (nombres et bornes) et leurs taux d'impositions. On pourra notamment expérimenter la dernière tranche à 0% et 100%.