

PENDANT TOUTE LA DUREE DU DEVOIR, L'UTILISATION DE DOCUMENTS DE COURS  
ET LA CONSULTATION D'INTERNET SONT STRICTEMENT INTERDITES.

### Première partie : travail sur feuille (10 points)

#### Exercice 1 : (7 points)

On considère le code suivant :

```

1 nombre = 0
2 produit = 1
3 while produit < 100:
4     nombre = nombre + 1
5     if nombre >= 3:
6         produit = produit * nombre * nombre
7     else:
8         produit = produit * nombre
9 print(nombre)
10 print(produit)

```

1. Dresser la liste des variables utilisées.
2. Repérer l'ensemble des affectations, en indiquant à chaque fois le numéro de ligne.
3. Repérer les instructions conditionnelles, en indiquant à chaque fois le numéro de ligne de début et de fin.
4. Repérer les boucles, en indiquant à chaque fois le numéro de ligne de début et de fin.
5. Compléter les cases blanches du tableau figurant en annexe et représentant l'état de la mémoire après l'exécution de chaque ligne de l'algorithme.
6. Quelles valeurs sont affichées à l'écran à la fin de l'exécution ?

#### Exercice 2 : (3 points)

Voici les règles d'un petit jeu :

- L'ordinateur lance 9 fois de suite deux dés équilibrés, et il compte le nombre de fois où la somme des deux dés vaut 7.
- Si au moins deux sommes sont égales à 7, le joueur a gagné. Dans le cas contraire, le joueur a perdu.

1. Parmi les instructions Python suivantes, laquelle permet de simuler le lancer d'un dé équilibré ? Justifier.

Réponse **A** : `randint(1, 2, 3, 4, 5, 6)`

Réponse **B** : `randint(0, 1)`

Réponse **C** : `randint(1, 6)`

Réponse **D** : `randint(de)`

2. À partir des lignes de code ci-dessous, reconstituer en annexe l'algorithme qui simule une partie de ce jeu, en prenant soin d'indenter correctement le code.

<code>de_1 = randint(.....)</code>	<code>cpt = 0</code>	<code>cpt = cpt + 1</code>	<code>else:</code>
<code>if de_1 + de_2 == 7:</code>	<code>if cpt &gt;= 2:</code>	<code>print("Bravo !")</code>	<code>for k in range(9):</code>
<code>de_2 = randint(.....)</code>	<code>print("Perdu !")</code>		

3. Écrire un programme qui simule plusieurs parties de ce jeu et qui affiche à l'écran le nombre de défaites subies par le joueur avant sa première victoire.

## Seconde partie : travail sur ordinateur (10 points)

Copier sur le bureau le fichier `devoir1.ipynb` depuis le dossier `Devoir` du réseau.

Lorsque vous aurez terminé, vous renommerez votre fichier `nom_prenom.ipynb` et vous le déposerez dans le dossier `Rendu` du réseau. Attention, une fois déposé dans `Rendu`, votre travail n'est plus modifiable.

NOM :

Annexe à rendre avec la copie

Exercice 1, question 5 :

Écrire le nom des variables ici →	Valeur des variables		Tests
			Vrai ou Faux
nombre = 0	0	<i>aucune valeur</i>	
produit = 1	0	1	
while produit < 100:			Vrai
nombre = nombre + 1	1	1	
if nombre >= 3:			Faux
produit = produit * nombre * nombre	<i>ligne non exécutée</i>		
else:			
produit = produit * nombre	1	1	
while produit < 100:			
nombre = nombre + 1			
if nombre >= 3:			
produit = produit * nombre * nombre			
else:			
produit = produit * nombre			
while produit < 100:			
nombre = nombre + 1			
if nombre >= 3:			
produit = produit * nombre * nombre			
else:			
produit = produit * nombre			
while produit < 100:			
nombre = nombre + 1			
if nombre >= 3:			
produit = produit * nombre * nombre			
else:			
produit = produit * nombre			
while produit < 100:			
nombre = nombre + 1			
if nombre >= 3:			
produit = produit * nombre * nombre			
else:			
produit = produit * nombre			
print(nombre)			
print(produit)			

Exercice 2, question 2 :

Lignes de code à utiliser :

de_1 = randint(.....)	cpt = 0	cpt = cpt + 1	else:
if de_1 + de_2 == 7:	if cpt >= 2:	print("Bravo !")	for k in range(9):
de_2 = randint(.....)	print("Perdu !")		

Prendre soin d'indenter correctement son code !

Code reconstitué :

[illegible]