

Important :

Le candidat est appelé à créer, dans le répertoire **Bac2024**, un dossier de travail ayant comme nom son numéro d'inscription (6 chiffres) et dans lequel il doit enregistrer, au fur et à mesure, tous les fichiers solution à ce sujet.

Transformation Powertrain

On se propose de concevoir une interface graphique permettant de saisir deux entiers **N** ($200 \leq N \leq 999999$) et **M** ($3 \leq M \leq 10$), puis d'afficher la **transformation Powertrain** de l'entier **N** ainsi que celles des **M** entiers consécutifs qui le suivent.

La **transformation Powertrain** d'un entier **X** consiste à générer un entier **Y** à partir des chiffres de **X**, en multipliant dans l'ordre chaque chiffre de rang impair élevé à la puissance du chiffre de rang pair.

Il est à noter que :

- Si le nombre de chiffres de **X** est impair alors le dernier chiffre est élevé à la puissance **1**.
- Si un chiffre de rang impair est égal à **zéro** et il est suivi d'un **zéro** alors la puissance **0⁰** sera remplacée par **1**.

Exemples :

- Pour **X = 523**, le nombre généré est **Y = $5^2 \times 3^1 = 25 \times 3 = 75$**
- Pour **X = 160071**, le nombre généré est **Y = $1^6 \times 0^0 \times 7^1 = 1 \times 1 \times 7 = 7$**

Travail demandé

1. Créer l'interface graphique illustrée dans la figure **Fig1** et l'enregistrer sous le nom **InterfacePowertrain**. Cette interface contient les éléments suivants :

- Un label contenant le texte "**Transformation Powertrain**",
- Un label contenant le texte "**N=**",
- Une zone de saisie pour la saisie d'un entier **N**,
- Un label contenant le texte "**M=**",
- Une zone de saisie pour la saisie d'un entier **M**,
- Un label pour afficher un premier message,
- Un label pour afficher un deuxième message,
- Un bouton intitulé "**Transformer**".

Transformation Powertrain	
N= <input type="text"/>	M= <input type="text"/>
Transformer	

Fig 1

2. Créer un programme en Python et l'enregistrer sous le nom **Powertrain** dans lequel on demande :

- a. d'implémenter l'algorithme suivant de la fonction **Puissance(A, B)** qui retourne la valeur d'un entier **A** élevé à la puissance **B** (notée **A^B** avec **A** et **B** deux entiers positifs).

Fonction Puissance(A, B : Entier) : Entier

DEBUT

P \leftarrow 1

Pour K de 1 à B Faire

P \leftarrow **P * A**

Fin Pour

Retourner P

FIN

T.D.O.L

Objet	Type/Nature
P, K	Entier

- b. de développer une fonction nommée **Transformer(X)** qui exploite la fonction **Puissance** pour déterminer la **transformation Powertrain** de l'entier **X**.
- c. de développer une fonction nommée **Chercher(N, M)**, qui exploite la fonction **Transformer** pour former une chaîne de caractères contenant la transformation **Powertrain** de l'entier **N** ainsi que celles des **M** entiers consécutifs qui le suivent séparés par le caractère "-" (voir **Fig3**).
- d. de développer un module **Play**, qui s'exécute suite à un clic sur le bouton "**Transformer**" permettant :
 - de récupérer les valeurs des deux entiers saisis **N** et **M** et de s'assurer de leurs validités afin d'afficher le message adéquat via le **label** dédié à l'affichage, comme illustré dans la figure **Fig2**,
 - d'exploiter la fonction **Chercher** afin d'afficher les messages adéquats via les **labels** dédiés à l'affichage, comme illustré dans la figure **Fig3**.
- e. d'exploiter l'annexe présentée ci-dessous tout en apportant les modifications nécessaires à l'intégration de l'interface graphique **InterfacePowertrain**.

Transformation Powertrain

N= **M=**

Transformer

Veuillez respecter $200 \leq N \leq 999999$ et $3 \leq M \leq 10$

Fig 2

Transformation Powertrain

N= **M=**

Transformer

La transformation de 523 et les 5 entiers suivants sont :

75-100-125-150-175-200

Fig 3

Annexe

```
from PyQt5.uic import loadUi
from PyQt5.QtWidgets import QApplication
.....
.....
app = QApplication([])
windows = loadUi ("Nom_Interface.ui")
windows.show()
windows.Nom_Bouton.clicked.connect (Nom_Module)
app.exec_()
```

Grille d'évaluation

Tâches	Nombre de points
1. Création de l'interface InterfacePowertrain .	3
2. Création du programme Powertrain .	17 =
a. Implémentation de la fonction Puissance .	3
b. Développement de la fonction Transformer .	5.5
c. Développement de la fonction Chercher .	3.5
d. Développement du module Play .	3.5
e. Exploitation de l'annexe.	1.5