

**Important :**

Le candidat est appelé à créer, dans le répertoire **BAC2025**, 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.

## Nombres ronds

On se propose de concevoir une interface graphique permettant de saisir deux entiers **A** et **B** tels que  $(10 \leq A \leq 30)$  et  $(A < B \leq 99)$ , puis d'afficher tous les nombres ronds compris entre **A** et **B**.

Un nombre est dit **rond** si sa conversion à la base 2 (base binaire) contient autant de 1 que de 0 (le nombre des 0 égal à celui des 1).

**Exemples :**

- Pour **A = 20 et B = 40** :

Le programme affiche : **Les nombres ronds sont : 35-37-38**

En effet, - la conversion binaire de  $35_{(10)}$  vaut  $100011_{(2)}$  qui contient autant de 1 que de 0.

- la conversion binaire de  $37_{(10)}$  vaut  $100101_{(2)}$  qui contient autant de 1 que de 0.
- la conversion binaire de  $38_{(10)}$  vaut  $100110_{(2)}$  qui contient autant de 1 que de 0.

- Pour **A = 15 et B = 30** :

Il n'existe aucun nombre rond compris entre **15** et **30**, d'où :

Le programme affiche : **Aucun nombre rond**

**Travail demandé**

- 1) Créer l'interface graphique illustrée dans la figure **Fig-1** et l'enregistrer sous le nom **Interface**.

Cette interface contient les éléments suivants :

- Un label contenant le texte : "**Nombres ronds**".
- Un label contenant le texte : "**A =**".
- Une zone de saisie pour la saisie d'un entier **A**.
- Un label contenant le texte : "**B =**".
- Une zone de saisie pour la saisie d'un entier **B**.
- Un bouton intitulé "**Afficher**".
- Un label dédié à l'affichage.

L'interface graphique "Nombres ronds" comprend les éléments suivants :

- Un label "Nombres ronds" en gras au sommet.
- Deux étiquettes "A =" et "B =" suivies de zones de saisie respectives.
- Un bouton "Afficher" en bas à droite.

**Fig-1**

- 2) Créer un programme python et l'enregistrer sous le nom **NombreRond**, dans lequel on demande :
- d'implémenter l'algorithme suivant de la fonction **Conv\_binaire** qui permet de convertir un entier **N** en binaire.

**Fonction Conv\_binaire (N : Entier) : Chaîne de caractères**

**DEBUT**

**Si (N=0) Alors**

ch  $\leftarrow$  "0"

**Sinon**

ch  $\leftarrow$  ""

**Tant que N  $\neq$  0 Faire**

r  $\leftarrow$  N Mod 2

N  $\leftarrow$  N Div 2

ch  $\leftarrow$  Convch ( r ) + ch

**Fin Tant que**

**Retourner ch**

**FIN**

**T.D.O.L**

Objet	Type/Nature
ch	Chaîne de caractères
r	Entier

- de développer une fonction **Verif(N)** qui permet de vérifier si un entier **N** est rond, en utilisant la fonction **Conv\_binaire** précédente.
- de développer une fonction **Ronds (A, B)** pour former une chaîne de caractères contenant les nombres ronds, de l'intervalle [A..B], séparés par le caractère "-", en utilisant la fonction **Verif**.
- de développer un module **Play**, qui s'exécute suite au clic sur le bouton "**Afficher**", permettant :
  - de récupérer les valeurs saisies des entiers **A** et **B**, de s'assurer de leurs validité et d'afficher, le cas échéant, le message adéquat via le label dédié à l'affichage, comme illustré dans les figures **Fig-2** et **Fig-3**.
  - d'exploiter la fonction **Ronds** afin d'afficher le message adéquat via le label dédié à l'affichage, comme illustré dans les figures **Fig-4** et **Fig-5**.
- d'exploiter l'annexe présentée ci-après tout en apportant les modifications nécessaires à l'intégration de l'interface graphique **Interface**.

**Nombres ronds**

A =   
 B =

**Afficher**

A doit être dans [10..30] !

**Fig-2**

**Nombres ronds**

A =   
 B =

**Afficher**

B doit être dans [20..99] !

**Fig-3**

**Nombres ronds**

A =   
 B =

**Afficher**

Les nombres ronds sont: 35-37-38

**Fig-4**

**Nombres ronds**

A =   
 B =

**Afficher**

Aucun nombre rond

**Fig-5**

## 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 <b>Interface</b> .	<b>3</b>
2) Création du programme <b>Nombre_Rond</b> .	<b>17 =</b>
a) Implémentation de la fonction <b>Conv_binaire</b>	<b>4</b>
b) Développement de la fonction <b>Verif</b>	<b>3</b>
c) Développement de la fonction <b>Ronds</b>	<b>2.5</b>
d) Développement du module <b>Play</b> .	<b>6</b>
e) Exploitation de l'annexe.	<b>1.5</b>