

RÉPUBLIQUE TUNISIENNE ----- MINISTÈRE DE L'ÉDUCATION	EXAMEN DU BACCALAURÉAT SESSION 2026	
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Sections : Mathématiques, Sciences expérimentales et Sciences techniques	
	Coefficient de l'épreuve : 0.5	Durée : 1h

Important

Le candidat est appelé à créer dans le répertoire **Bac2026**, 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.

Communication dans un réseau

Dans un réseau local, les machines communiquent entre elles grâce à des adresses IP uniques générées automatiquement par un serveur **DHCP** (Dynamic Host Configuration Protocol). Chaque adresse **IP** (IPv4), est de la forme **W.X.Y.Z** où **W**, **X**, **Y** et **Z** sont des nombres de **0** à **255** séparés par des points.

Pour mieux gérer la configuration d'un réseau local, l'administrateur informatique d'une société paramètre le serveur **DHCP** de sorte qu'il génère automatiquement des adresses IP distinctes de la forme **192.168.72.Z** où **Z** représente un nombre qui varie de **1** à **254**.

Dans ce contexte, on envisage de concevoir une application, enrichie par une interface graphique, qui simule l'attribution d'adresses IP aux machines d'un réseau local par un serveur **DHCP**, puis vérifie si deux machines d'adresses données **IP1** et **IP2** font partie de ce réseau local.

Cette application permet :

- de saisir un entier **N** ($5 \leq N \leq 200$) qui représente le nombre de machines du réseau local.
- de remplir un tableau **T** par **N** adresses IP distinctes générées aléatoirement de la forme **192.168.72.Z** (où **Z** représente un nombre qui varie de **1** à **254**),
- de saisir deux adresses IP distinctes notées **IP1** et **IP2** et d'afficher l'un des deux messages ci-dessous :
 - ✓ "**Communication possible**" lorsque les adresses **IP1** et **IP2** font partie du réseau local (**IP1** et **IP2** existent dans le tableau **T**).
 - ✓ "**Communication impossible**" dans le cas contraire.

Exemple :

Pour le tableau **T** suivant contenant **6** adresses **IP** distinctes générées aléatoirement par un serveur **DHCP** :

T	192.168.72.22	192.168.72.5	192.168.72.12	192.168.72.111	192.168.72.38	192.168.72.51
	0	1	2	3	4	5

Pour **IP1** = 192.168.72.5 et **IP2** = 192.168.72.51 le programme affiche "**Communication possible**" car **IP1** et **IP2** existent dans le tableau **T**.

Pour **IP1** = 192.168.72.55 et **IP2** = 192.168.72.51 le programme affiche "**Communication impossible**" car **IP1** n'existe pas dans le tableau **T**.

Travail demandé

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

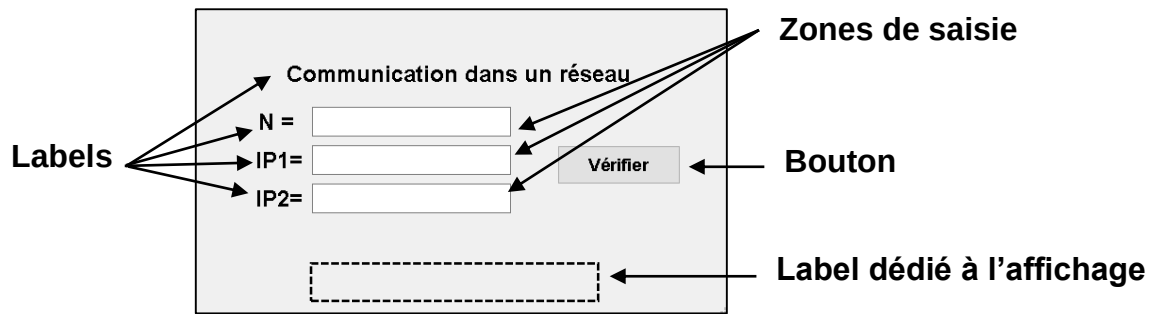


Fig-1

2. Créer un programme en python et l'enregistrer sous le nom **Réseau**, dans lequel on demande :

- a. D'implémenter l'algorithme suivant de la procédure **DHCP** permettant de remplir, aléatoirement, un tableau **T** par **N** adresses **IP** de la forme **192.168.72.Z** (où **Z** représente un nombre qui varie de **1** à **254**).

Procédure DHCP (@ T : TAB, N : Entier) DEBUT $T[0] \leftarrow "192.168.72." + \text{Convch}(\text{Aléa}(1, 254))$ Pour i de 1 à N-1 Faire Répéter $T[i] \leftarrow "192.168.72." + \text{Convch}(\text{Aléa}(1, 254))$ Jusqu'à Existe (T, $i-1$, $T[i]$) = Faux Fin Pour FIN	
Nouveau type TAB = Tableau de 200 Chaîne de caractères	
Objet	Type/Nature
i	Entier
Existe	Fonction

- b. D'écrire une fonction booléenne **Existe (T, K, CH)** qui permet de vérifier l'existence d'une chaîne de caractères **CH** dans un tableau **T** contenant **K** chaînes de caractères.
- c. D'écrire une fonction booléenne **Format (CH)** qui permet de vérifier si une chaîne de caractères **CH** est de la forme **192.168.72.Z** (**Z** représente un nombre qui varie de **1** à **254**).
- d. D'écrire un module **Play**, qui s'exécute suite au clic sur le bouton "**Vérifier**", permettant :
- ✓ de récupérer la valeur du nombre **N** saisi, puis de s'assurer de sa validité comme décrit précédemment et d'afficher, le cas échéant, le message adéquat via le label dédié à l'affichage, comme illustré dans la figure **Fig-2** et **Fig-3**,
 - ✓ de récupérer les valeurs des adresses **IP1** et **IP2**, puis de s'assurer :
 - de leurs validités, en utilisant la fonction **Format**, tout en affichant, le cas échéant, le message adéquat via le label dédié à l'affichage, comme illustré dans la figure **Fig-4**,
 - de leurs éventuelles communication, en utilisant les modules **DHCP** et **Existe**, tout en affichant le message adéquat via le label dédié à l'affichage, comme illustré dans les figures **Fig-5** et **Fig-6**.
- e. D'exploiter l'annexe présentée ci-après tout en apportant les modifications nécessaires à l'intégration de l'interface graphique **InterfaceRéseau**.

Communication dans un réseau

N =

IP1=

IP2=

Vérifier

Veuillez saisir un nombre !

Fig-2

Communication dans un réseau

N =

IP1=

IP2=

Vérifier

N doit être 5 et 200

Fig-3

Communication dans un réseau

N =

IP1=

IP2=

Vérifier

L'adresse IP doit être de la forme : 192.168.72.Z

Fig-4

Communication dans un réseau

N =

IP1=

IP2=

Vérifier

Communication possible

Fig-5

Communication dans un réseau

N =

IP1=

IP2=

Vérifier

Communication impossible

Fig- 6

Annexe

```
from PyQt5.uic import loadUi
from PyQt5.QtWidgets import QApplication
from numpy import array
import random
T=array([str]*200)
.....
.....
app = QApplication([])
windows = loadUi ("Nom_Interface.ui")
windows.show()
windows.Nom_Bouton.clicked.connect (Nom_Module)
app.exec_()
```

Grille d'évaluation

Tâche	Nombre de points
1. Création de l'interface InterfaceRéseau .	3
2. Création du programme Réseau .	17 =
a) Implémentation du module DHCP .	3
b) Développement du module Existe	3
c) Développement du module Format	4
d) Développement du module Play .	6
e) Exploitation de l'annexe.	1