Nom :

Prénom :

1ère BAC Pro CIEL	Manipulation des listes	exos Année 2024/2025

Exercice 1 : Adresse IP autorisée

Un technicien réseau doit vérifier les adresses IP autorisées sur un switch administrable.

Créer une liste contenant les adresses IP suivantes :

```
192.168.1.10
192.168.1.11
192.168.1.12
192.168.1.13
```

Afficher toute la liste

Afficher uniquement la première IP

Afficher uniquement la dernière IP

Exercice 2 : Ajout d'un équipement réseau

Un nouveau poste informatique vient d'être ajouté dans le réseau.

Créer la liste suivante :

```
["PC-01","PC-02","PC-03"]
```

Ajouter « PC-04 » à la liste

Afficher la liste

Afficher le nombre total d'équipement avec len()

Exercice 3 : Suppression d'un utilisateur bloqué

Un compte utilisateur compromis doit être retiré d'un système.

Créer la liste :

```
["admin","technicien","invite","stagiaire"]
```

Supprimer « invite »

Afficher la nouvelle liste

Vérifier si « invite » est encore présent

Exercice 4 : Analyse d'un scan réseau

Après un scan réseau, plusieurs ports ouverts ont été détectés.

Créer la liste :

```
[80,443,22,21,8080]
```

Parcourir la liste et afficher :

```
Port détecté : x
```

Avec x le numéro de port

Afficher le nombre total de ports détectés

Exercice 5 : Détection d'une adresse IP suspecte

Un pare-feu surveille des connexions réseau.

Créer la liste :

```
["10.0.0.5","192.168.1.25","172.16.0.8"]
```

Demander à l'utilisateur une adresse IP.

Le programme doit :

- ⇨ Indiquer si l'adresse est présente
- ⇨ Afficher :
  - « IP détectée » : si l'IP se trouve dans la liste
  - « IP inconnue » : si l'IP ne se trouve pas dans la liste

Exercice 6 : Journal des tentatives de connexion

Un serveur conserve les tentatives de connexion d'utilisateurs.

Créer une liste vide.

Le programme doit :

- ⇨ Demander 5 noms d'utilisateurs
- ⇨ Les ajouter dans une liste
- ⇨ Afficher ensuite tous les utilisateurs enregistrés tel que :

```
Connexion détectée : admin

Connexion détectée : test
```

Exercice 7 : Tri des équipements réseau

Un administrateur souhaite trier des équipements réseau.

Créer la liste :

```
["PC-12","PC-02","PC-25","PC-01"]
```

Afficher la liste avant tri

Trier la liste

Afficher la liste après tri

Exercice 8 : Supervision de température serveur

Une salle serveur surveille les températures de plusieurs baies informatiques.

Créer la liste

```
[32,35,31,40,29,45]
```

Afficher toutes les températures

Afficher uniquement les températures supérieures à 35°C

Afficher combien de serveurs dépassent 35°C

Exercice 9 : Analyse d'alertes cybersécurité

Un logiciel SOC (Security Operation Center) enregistre des alertes de sécurité

Créer la liste :

```
["LOW","HIGH","MEDIUM","HIGH","LOW","CRITICAL"]
```

Le programme doit :

- ⇨ Compter le nombre d'alertes « HIGH »
- ⇨ Vérifier si « CRITICAL » est présent et afficher « ALERTE CRITIQUE DETECTEE » si présent

Exercice 10 : Blacklist d'adresse IP

Un pare-feu doit gérer automatiquement une blacklist.

Créer une liste vide appelée blacklist

Le programme doit :

- ⇨ Demander 5 adresses IP à bloquer
- ⇨ Les stocker dans la liste blacklist
- ⇨ Afficher la blacklist complète
- ⇨ Demander une IP à supprimer
- ⇨ Supprimer cette IP si elle existe
- ⇨ Réafficher la blacklist

Exercice 11 : Analyse simplifiée de logs réseau

Un administrateur reçoit une liste de connexions réseau

Créer la liste :

```
["OK","OK","ECHEC","OK","ECHEC","ECHEC","OK"]
```

Le programme doit :

- ⇨ Compter le nombre de connexion réussies
- ⇨ Compter le nombre d'échecs
- ⇨ Afficher un message d'alerte si le nombre d'échecs dépasse 2

Exercice 12 : Mini projet – Surveillance d'un serveur

Un serveur de supervision reçoit les états des services réseau.

Créer la liste :

```
["HTTP","SSH","FTP","DNS"]
```

Le programme doit :

- ⇨ Afficher tous les services
- ⇨ Ajouter « SMTP »
- ⇨ Supprimer « FTP »
- ⇨ Vérifier si « SSH » est actif
- ⇨ Afficher le nombre total de services
- ⇨ Trier les services
- ⇨ Afficher la liste finale

Exercice Finale : Analyse log IP

Un serveur de supervision a généré un fichier de logs contenant des tentatives de connexion réseau.

Chaque ligne du fichier est sous la forme :

```
adresse_ip ;port ;etat
```

Exemple :

```
192.168.1.25;21;OK
```

```
192.168.1.15;3389;ECHEC
```

Le fichier contient plusieurs connexions avec différentes adresses IP, ports et résultats de connexion.

Le programme devra utiliser deux listes : ip et echec. Ces deux listes devront fonctionner avec des index parallèles.

Etape 1 :

Créer les deux listes vides.

Etape 2 :

Parcourir les logs. Pour chaque ligne de logs :

Séparer les informations avec split.

Récupérer l'adresse IP, le port et le statut.

Etape 3 :

Ajouter les nouvelles IP. Pour chaque adresse IP trouvée dans le fichier :

Si l'adresse IP n'est pas déjà présente dans liste ip, l'ajouter et ajouter également 0 dans la liste echec

Etape 4 :

Compter les échecs. Si l'état de la ligne est « ECHEC » :

- ⇨ Trouver l'index de l'adresse IP dans la liste ip
- ⇨ Ajouter 1 au compteur correspondant dans la liste echec

Etape 5 :

A la fin du programme, afficher le nombre d'échecs pour chaque adresse IP.

Etape 6 :

Une adresse IP est considérée comme suspecte si elle a généré plus de 3 échecs.

Afficher la liste des adresses IP suspectes.

Exemple :

Liste ip :

```
["192.168.1.57","192.168.1.31","192.168.1.63"]
```

Liste erreur :

```
[1,9,3]
```

Ces deux listes indiquent que :

L'adresse IP 192.168.1.57 a eu 1 echec

L'adresse IP 192.168.1.31 a eu 9 echec

L'adresse IP 192.168.1.63 a eu 3 echec

A la fin, le programme devra afficher pour chaque adresse IP, le nombre d'échecs :

```
Adresse IP 192.168.1.57 : 1 échecs
Adresse IP 192.168.1.31 : 9 échecs
Adresse IP 192.168.1.63 : 3 échecs
```

Et enfin un rapport indiquant les adresses IP suspectes (les adresses IP ayant eu plus de 3 échecs :

```
*** Adresses IP suspectes ***
192.168.1.31
```