

TRAVAUX PRATIQUES
La programmation en Python
T.P. n°5 : Les fonctions

BTS CIEL
2 ^{ème} année
Page 1 sur 2

Exercice 1 : Gestion des trames GPS avec coordonnées erronées

Dans un réseau IoT, les trames GPS reçues de plusieurs appareils peuvent parfois contenir des erreurs de transmission. Écrivez une fonction qui prend en entrée des coordonnées GPS (latitude et longitude) et vérifie si elles sont valides (par exemple, $-90 \leq \text{latitude} \leq 90$ et $-180 \leq \text{longitude} \leq 180$). Si une coordonnée est invalide, l'appareil doit être marqué pour vérification.

Contraintes :

- Utilisez des boucles pour vérifier chaque paire de coordonnées reçues.

Exercice 2 : Surveillance et contrôle du débit de bande passante

Écrivez une fonction qui simule le contrôle de l'utilisation de la bande passante d'un appareil IoT connecté à un réseau. La fonction prend en entrée un débit réseau simulé et un seuil de bande passante autorisé. Si le débit réseau dépasse ce seuil pendant un certain nombre d'itérations (ex. : 3 fois consécutivement), une alerte est envoyée pour limiter ou couper la connexion de l'appareil.

Contraintes :

- Simulez le débit réseau avec une boucle qui génère des valeurs aléatoires.
- Utilisez une condition pour vérifier si le seuil est dépassé plusieurs fois de suite avant de déclencher l'alerte.

Résultat souhaité :

```
Iteration 1: Débit réseau 147 Mbps dépasse le seuil de 100 Mbps.
Iteration 2: Débit réseau 61 Mbps est sous contrôle.
Iteration 3: Débit réseau 142 Mbps dépasse le seuil de 100 Mbps.
Iteration 4: Débit réseau 67 Mbps est sous contrôle.
Iteration 5: Débit réseau 88 Mbps est sous contrôle.
Iteration 6: Débit réseau 79 Mbps est sous contrôle.
Iteration 7: Débit réseau 104 Mbps dépasse le seuil de 100 Mbps.
Iteration 8: Débit réseau 94 Mbps est sous contrôle.
Iteration 9: Débit réseau 74 Mbps est sous contrôle.
Iteration 10: Débit réseau 132 Mbps dépasse le seuil de 100 Mbps.
```

TRAVAUX PRATIQUES
La programmation en Python
T.P. n°7 : Les fonctions

BTS CIEL
2 ^{ème} année
Page 2 sur 2

Exercice 3 : Analyse des logs de connexion SSH

Écrivez une fonction qui simule l'analyse des logs de connexions SSH pour détecter des tentatives de connexion échouées. La fonction doit prendre en entrée le nombre de tentatives de connexion (par exemple 10), puis simuler une série de connexions réussies ou échouées. Si trois tentatives échouées sont détectées consécutivement, une alerte doit être déclenchée pour bloquer l'utilisateur.

Contraintes :

- Utilisez une boucle pour simuler les tentatives de connexion SSH.
- Utilisez des conditions pour détecter trois échecs consécutifs.

Résultats souhaités :

```
Tentative 1: Connexion échouée.
Tentative 2: Connexion échouée.
Tentative 3: Connexion réussie.
Tentative 4: Connexion échouée.
Tentative 5: Connexion réussie.
Tentative 6: Connexion échouée.
Tentative 7: Connexion échouée.
Tentative 8: Connexion échouée.
Alerte : Trois connexions échouées consécutives. Compte bloqué.
```

Exercice 4 : Système de vérification de mot de passe

Écrivez une fonction qui permet à un utilisateur d'essayer de se connecter à un système en saisissant un mot de passe. La fonction doit :

1. Prendre un mot de passe défini à l'avance (comme "admin123") et permettre à l'utilisateur d'entrer son mot de passe.
2. Limiter à **3 tentatives**. Si l'utilisateur entre un mauvais mot de passe trois fois de suite, une alerte est affichée et l'accès est bloqué.
3. Si le mot de passe est correct avant d'atteindre la troisième tentative, afficher un message indiquant que l'accès est autorisé.