Contrôler les sauvegardes

Module 11 • Comme au CCF 1 • Entraînement de 60 minutes • BTS SIO1

Nom : _____ Date : _____

Chaque matin, le service informatique reçoit une liste des états de sauvegarde des postes. Vous devez compter les sauvegardes en erreur et les sauvegardes absentes, puis décider si une intervention est nécessaire.

État	Signification
"OK"	La sauvegarde a réussi.
"ERREUR"	La sauvegarde a été lancée, mais a échoué.
"ABSENT"	Aucune sauvegarde n'a été reçue.

Les données contiennent uniquement ces trois textes, écrits exactement ainsi. L'ordre de la liste correspond à l'ordre des postes. La liste peut être vide. Dans cet entraînement, une liste vide ne déclenche pas d'intervention : aucun poste n'est à traiter.

Mémo autorisé. Aides graduées sur un document séparé. Cherchez avant de les consulter, puis notez le niveau utilisé. Les questions 1 à 6 constituent le parcours d'une heure ; le défi de la page 4 est facultatif. Écrivez les programmes au dos ou dans votre éditeur si nécessaire.

1 Comprendre les données 7 min

```
etats = ["OK", "ERREUR", "ABSENT", "OK", "ERREUR"]
```

1a. Nombre d'éléments : _____. Valeur de etats[2] : _____.

1b. Indices des deux sauvegardes "ERREUR" : _____ et _____.

1c. Sans programme : nombre d'erreurs _____ ; nombre d'absences _____.

Rappel : i est un indice ; etats[i] est l'état à cette position. range(len(etats)) parcourt les indices de 0 à len(etats) - 1 ; si la liste est vide, aucun passage.

2 Analyser une fonction 10 min

```
def compter_erreurs(etats):
    nombre = 0
    for i in range(len(etats)):
        if etats[i] == "ERREUR":
            nombre = nombre + 1
    return nombre
```

2a. Complétez la trace avec la liste ci-dessus, sans exécuter la fonction.

Indice i	0	1	2	3	4
État etats[i]	OK	ERREUR	ABSENT	OK	ERREUR
nombre après ce passage					

2b. Valeur retournée : _____. Expliquez pourquoi "ABSENT" n'est pas compté.

2c. Pourquoi nombre est-il initialisé avant la boucle et return placé après ?

.....

.....

Compléter traduire et réparer

3 Compter les sauvegardes absentes 15 min

3a. Complétez le pseudo-code pour renvoyer uniquement le nombre d'états "ABSENT". La fonction doit aussi fonctionner avec une liste vide.

```
Fonction compter_absentes(etats : liste) : entier
    nombre ← ______
    Pour i de 0 à ______ Faire
        Si ______ Alors
            nombre ← ______
        Fin Si
    Fin Pour
    Retourner ______
Fin Fonction
```

3b. Traduisez en Python avec for i in range(len(etats)), puis écrivez un appel qui affiche le résultat pour la liste de la page 1. Définir une fonction ne suffit pas à l'exécuter.

.....

.....

.....

.....

.....

.....

3c. Prévoyez puis vérifiez : liste de la page 1 → _____ ; [] → _____ ;
["ABSENT", "ABSENT"] → _____. Une différence constatée : _____

4 Réparer une règle mal traduite 10 min

Une autre version de compter_erreurs est proposée. Elle doit compter uniquement les états "ERREUR". Pourtant, elle ne respecte pas toujours cette règle.

```
def compter_erreurs(etats):
    nombre = 0
    for i in range(len(etats)):
        if etats[i] != "OK":
            nombre = nombre + 1
    return nombre
```

4a. Avec ["OK", "ABSENT"], résultat attendu : _____ ; résultat réel : _____.
4b. Expliquez ce que cette version compte réellement, puis corrigez une ligne.

.....

.....

4c. Proposez un autre test qui révèle l'erreur, avec son résultat attendu.
Un test doit permettre de distinguer le programme faux du programme corrigé.

.....

.....

Décider et vérifier

5 Créer une fonction de décision 12 min

Une intervention est nécessaire dès qu'au moins une sauvegarde est en erreur ou absente. Écrivez une function intervention_necessaire(etats), qui renvoie un booléen : Vrai ou Faux en pseudo-code, True ou False en Python.

5a. Écrivez le pseudo-code, puis Python. Réutilisez les fonctions compter_erreurs et compter_absentes, considérées comme correctes même si vous n'avez pas terminé les questions précédentes. Une liste contenant uniquement "OK", ou une liste vide, doit donner Faux.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

5b. Écrivez l'appel qui affiche le résultat de la décision pour la liste de la page 1. Résultat attendu : _____.

.....

.....

6 Choisir des tests et faire le point 6 min

6a. Avant d'exécuter votre fonction, choisissez deux tests supplémentaires : un qui doit renvoyer True et un qui doit renvoyer False. Évitez de reprendre la liste de la page 1.

Liste choisie	Résultat attendu	Résultat vérifié

6b. Expliquez à un voisin pourquoi une liste vide doit donner False dans ce sujet. Si un résultat diffère, revenez à la règle avant de modifier le code.

Mon aide la plus élevée : aucune / 1 / 2 / 3.
Je sais expliquer une correction : seul / avec une trace / avec aide.
La question qui m'a le plus fait hésiter : _____. Pourquoi ? _____

Défi sur les postes à vérifier

Environ 10 minutes • À commencer seulement après le parcours commun

On veut désormais renvoyer une liste des indices des postes dont la sauvegarde est "ERREUR" ou "ABSENT". Pour la liste de la page 1, le résultat doit être [1, 2, 4]. On attend des indices commençant à 0, pas des numéros commençant à 1.

Compléter puis traduire

```
Fonction indices_a_verifier(etats : liste) : liste
    indices ← ______
    Pour i de 0 à Longueur(etats) - 1 Faire
        Si ______ Alors
            Ajouter ______ à indices
        Fin Si
    Fin Pour
    Retourner ______
Fin Fonction
```

Traduisez en Python en utilisant for i in range(...). Prévoyez puis vérifiez les résultats pour [], ["OK"] et ["ABSENT"]. Les aides du défi sont sur le document séparé.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Pourquoi ajoute-t-on i à la liste de résultat, plutôt que etats[i] ?

.....

.....

.....