

“

**ENSEIGNEMENT ET
INNOVATION #5**



de la Pédagogie

**La classe inversée pour
apprendre la programmation**

Avec Dr Sébastien HOARAU, Maître de Conférences en Informatique
Chercheur au Laboratoire d'Informatique et de Mathématiques de
l'Université de La Réunion

Corrigé des exercices sur les séquences

5.2 adn

Il s'agit d'écrire une fonction `est_adn` qui prend une chaîne de caractères `s` en paramètres et qui renvoie `True` si `s` ne contient que les caractères `A`, `C`, `G` ou `T` et `False` sinon.

L'algorithme mis en place : on parcourt chacune des lettres (on reconnaît ici le schéma d'une boucle `for` : en Python une chaîne de caractères se comporte comme un itérable de caractères) et on va tester qu'on a bien `A`, `C`, `G` ou `T`.

Première version

```
[ ]: def est_adn_v1(brin):  
    for c in brin:  
        if c in 'ACGT':  
            pass  
        else:  
            return False  
    return True
```

On constate que lorsque le caractère est bien par `'ACGT'`, on n'a rien à faire à part passer au caractère suivant... le bon test est donc l'opposé : si le caractère **n'est pas** dans `'ACGT'` alors c'est que le `brin` n'est pas *adn* et je dois renvoyer `False` :

Deuxième version

Boucle bornée

La boucle bornée désigne l'instruction `for` qui exécute n fois une séquence d'instructions. Cet entier n peut être nul mais est toujours positif et ne change pas. En Python, comme dans bien d'autres langages, la boucle bornée est mise en œuvre par l'instruction `for`.

↑ Retour en haut de la page

Fiches cours

[Notions transversales de programmation \(rappel 2nde\)](#)

Représentation des données : entiers et flottants

Représentation des données : booléens

Représentation des données : p-uplets, p-uplets nommés

Représentation des données : tableaux

Représentation des données : dictionnaires

Traitement de données en tables

Spécification, documentation et tests

Recherche d'un élément dans un tableau : parcours séquentiel et recherche dichotomique

Table des matières

[Introduction](#)

[Valeurs, variables, affectation](#)

[Séquences d'instructions](#)

[Instruction conditionnelle](#)

[Boucle bornée](#)

[Boucle non bornée](#)

[Fonctions prédéfinies et appels de fonctions](#)

[Définitions de fonctions](#)

Syntaxe de la boucle bornée de Python

```
for i in range(n):  
    # séquence d'instructions  
    # aussi appelée corps de la boucle  
# suite du programme
```

Sémantique de boucle bornée

La variable `i` va prendre tour à tour les valeurs 0 à $n - 1$. À chaque nouvelle affectation de `i`, le corps de la boucle est exécuté.

À faire vous-même...

Jouez avec le `for` ci-dessous ; vous pouvez aussi ajouter des instructions : par exemple, ajoutez un `i`, dans le `print`, avant le `'Hello'`.

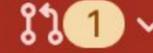
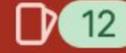
```
1 for i in range(3):  
2     print('Hello')  
3
```

Your project is no longer receiving GitLab Ultimate benefits as of 2022-07-01. As notified in-app previously, public open source projects on the Free tier can apply to the GitLab for Open Source Program to receive GitLab Ultimate benefits. Please refer to the FAQ for more details.

Merge branch 'main' of https://gitlab.com/univ-reunion/didactique-informatique into main
Nathan PRIGENT authored 1 day ago

main didactique-informatique / docs / fiches_cours / + History Find file Web IDE Clone

Table with 3 columns: Name, Last commit, Last update. Rows include files like Fiche_1_programmation.md, Fiche_2_representation_nombres.md, etc.



D

```
311
312 ## Boucle bornée
313
314 La boucle bornée désigne l'instruction qui permet d'effectuer $n$ fois une séquence d'instruction
315
316 !!! tip "Syntaxe de la boucle bornée de Python"
317
318     ```python
319     for i in range(n):
320         # séquence d'instructions
321         # aussi appelée corps de la boucle
322     # suite du programme
323     ```
324
325 !!! note "Sémantique de boucle bornée"
326
327     La variable `i` va prendre tour à tour les valeurs $0$ à $n-1$. À chaque nouvelle affectation
328
329 ???+ exemple "À faire vous-même..."
330
331     Jouez avec le `for` ci-dessous ; vous pouvez aussi ajouter des instructions : par exemple, aj
332
333     {{ IDE('01/ex_for') }}
334
335 ???+ warning "Remarque"
```

Accueil

Ressources proposées par e-nsi

Origine du groupe

Le groupe *e-nsi* s'est formé au printemps 2022 dans le cadre du Forum Numérique et Sciences Informatiques.

Regroupant des professeurs d'informatique du secondaire et du supérieur, il vise à produire **collectivement** des ressources pédagogiques sur le thème des sciences informatiques.

Les ressources produites :

- sont travaillées et critiquées à plusieurs par des membres soucieux de qualité.
- sont utilisables sans inscription, sans installation, dans le respect total du RGPD.
- sont facilement ré-exploitable, tout est écrit avec du simple Markdown.
- sont bien indexées, avec des tags, des parcours.

Un puissant moteur de recherche vous aidera sur chaque site

Forum Numérique et Sciences Informatiques
aeif Inria LearningLab FUN MOOC

toutes les catégories toutes les étiquettes Catégories Récents

Accueil

Catégorie	Sujets
Privé : espace modérateurs	25
Échanges entre modérateurs. Également plateforme de tests.	
Salon de discussion	304
On s'y présente, on y discute de l'utilisation de ce site, mais surtout de tout et de rien, et parfois d'autre chose.	
Au sujet de ce forum Café Présentez-vous ! Bonnes pratiques FUN Mooc Les fondamentaux FUN Mooc La pratique	
Faire communauté	66
Exprimez votre créativité. Proposer des exercices nouveaux, de types variés.	