

Documentation du projet “Mot1”

Binome 1 Correction

9 juin 2025

1 Introduction

Ce projet Python a pour but de manipuler des mots définis sur des alphabets personnalisables. Il se compose de trois fichiers principaux :

- `mots.py` : définit la classe `Mot1` pour manipuler les mots.
- `alpha.py` : permet de choisir et personnaliser un alphabet.
- `tests1.py` : script de test interactif pour manipuler un mot avec l’alphabet choisi.

Les modules sont conçus de manière à être indépendants et réutilisables.

2 Fichier `mots.py` — Classe `Mot1`

2.1 Description générale

La classe `Mot1` permet de créer un objet représentant un mot, de le nettoyer, de vérifier sa conformité à un alphabet donné, et d’effectuer diverses opérations : concaténation, adjonction de symboles, génération de sous-mots et de facteurs.

2.2 Constantes globales

- `SYMBOLE_MOT_VIDE = '&'` : représente le mot vide.
- `SYMBOLS_WHITESPACE` : ensemble des symboles blancs à supprimer lors du nettoyage du mot.

2.3 Méthodes de la classe `Mot1`

- `__init__(contenu : str, alphabet : Optional[Set[str]])`
Nettoie le contenu du mot en supprimant les symboles blancs. Vérifie que le contenu appartient à l’alphabet fourni. Permet d’interagir avec l’utilisateur pour confirmer le nettoyage.
- `__repr__()`
Affiche une représentation lisible de l’objet `Mot1`.
- `longueur()`
Retourne la longueur du mot (en ignorant les symboles blancs et `SYMBOLE_MOT_VIDE`).
- `adjonction_occurrence_droite(symbole)`
Ajoute un symbole à droite du mot, tout en respectant l’alphabet.
- `adjonction_occurrence_gauche(symbole)`
Ajoute un symbole à gauche du mot, tout en respectant l’alphabet.
- `concatenation(autre_mot, separateur = "")`
Concatène le mot courant avec un autre mot (`Mot1`). Les deux mots doivent partager le même alphabet.

- `liste_sous_mots(distinct = False, min_length = 1)`
Retourne la liste de tous les **sous-mots** possibles du mot : sous-séquences respectant l'ordre des lettres, mais non nécessairement contiguës.
- `liste_facteurs(distinct = False, min_length = 1)`
Retourne la liste de tous les **facteurs** du mot : sous-chaînes contiguës.

2.4 Résumé

Le fichier `mots.py` fournit un outil très souple pour manipuler des mots et explorer leurs propriétés formelles en lien avec la théorie des mots.

3 Fichier `alpha.py` — Gestion des alphabets

3.1 Description générale

Ce fichier fournit une fonction principale `choisir_alphabet` qui permet à l'utilisateur d'interagir avec un menu pour choisir, créer, combiner ou modifier un alphabet.

3.2 Constante `ALPHABETS_PREDEFINIS`

Contient plusieurs ensembles de symboles prédéfinis :

- `lettres` : lettres minuscules.
- `chiffres` : chiffres de 0 à 9.
- `lettres_majuscules` : lettres majuscules.
- `lettres_et_chiffres` : combinaison de lettres majuscules, minuscules et chiffres.

3.3 Fonction `choisir_alphabet()`

Permet à l'utilisateur de :

- Choisir un ou plusieurs alphabets existants.
- Créer un nouvel alphabet personnalisé.
- Combiner des alphabets.
- Modifier un alphabet (ajouter ou retirer des symboles).

Cette fonction retourne l'alphabet validé, qui sera utilisé pour créer des objets `Mot1`.

3.4 Résumé

Le fichier `alpha.py` fournit un outil très flexible pour la construction d'alphabets personnalisés. Il permet une interaction conviviale avec l'utilisateur.

4 Fichier `tests1.py` — Script de test interactif

4.1 Description générale

Ce fichier contient le **programme principal** du projet.

Il relie les deux autres fichiers en permettant :

- de choisir un alphabet via `choisir_alphabet()` ;
- de saisir un mot en respectant cet alphabet ;
- de créer un objet `Mot1` ;
- d'exécuter diverses opérations sur ce mot.

4.2 Fonction `main()`

Contient une boucle `while True` qui guide l'utilisateur pas à pas :

1. Choix de l'alphabet ;
2. Saisie du mot ;
3. Affichage du mot nettoyé et de son alphabet ;
4. Calcul et affichage de la longueur ;
5. Adjunction d'un symbole à gauche/droite ;
6. Concaténation avec un autre mot ;
7. Génération de la liste de facteurs ;
8. Génération de la liste de facteurs distincts.

4.3 Résumé

Le fichier `tests1.py` sert de démonstration interactive de la classe `Mot1` et du module `alpha.py`. Il permet de tester toutes les fonctionnalités de manière progressive.

5 Lien entre les fichiers

- `mots.py` fournit la classe centrale `Mot1`.
- `alpha.py` permet de préparer un alphabet conforme pour les mots.
- `tests1.py` utilise `Mot1` et `choisir_alphabet()` pour proposer une interface utilisateur complète.

Le script `tests1.py` dépend de `alpha.py` et `mots.py`. Les deux autres fichiers peuvent être réutilisés indépendamment dans d'autres projets.

6 Conclusion

Ce projet offre une architecture modulaire pour la manipulation des mots sur des alphabets personnalisés. La classe `Mot1` et le module `alpha` sont facilement extensibles.