

# Parenthésage correct

## Parenthésage correct

Avant d'évaluer une expression, on souhaite vérifier qu'elle est correctement parenthésée.

On appelle **expression correctement parenthésée** une expression dans laquelle, à chaque position, le nombre d'*ouvrants* d'une catégorie situés avant cette position est toujours supérieur ou égal au nombre de *fermants* de la même catégorie situés avant cette position.

Par exemple, l'expression " $3*(8+2)$ " est correctement parenthésée alors que l'expression " $\{8, 2\}$ " ne l'est pas.

On considère plusieurs catégories de **délimiteurs** :

- les parenthèses "(" et ")",
- les crochets "[" et "]",
- les accolades "{" et "}".

Ces délimiteurs sont de deux **genres** :

- certains sont *ouvrants*, comme "(", "[" et "{".
- les autres sont *fermants*, comme ")", "]" et "}".

Les différents délimiteurs sont regroupés dans un dictionnaire {**délimiteur**: (**genre**, **complémentaire**)} qui associe par exemple à la clé "]" le tuple ("fermant", "[").

## Objectif

On demande d'écrire la fonction `parenthesage_correct` qui prend en paramètre une chaîne de caractères **entree** et renvoyant **True** si elle est correctement parenthésée, **False** dans le cas contraire.

## Exemples

```
>>> parenthesage_correct("3*(8+2)")
```

```
True
>>> parenthesisage_correct("[3*(8+2)]")
True
>>> parenthesisage_correct("[{8, 2}]")
False
>>> parenthesisage_correct("() [{}]")
True
>>> parenthesisage_correct("() [{}]")
False
```