

2. Remplacer chaque élément de `lst` par le mot 'pair' ou 'impair' suivant que l'indice de l'élément est pair ou impair.

Exercice 6 *4 points*

On considère la liste

```
tab = ["un", "deux", "trois", "quatre", "cinq", "six", "sept", "huit", "neuf"]
```

1. Écrire une fonction `nb_voyelles` qui prend en paramètres une chaîne de caractères `mot` et qui renvoie le nombre de voyelles de `mot`.

2. Écrire un code permettant de construire une liste `new_tab` qui contient les éléments de `tab` ayant exactement 2 voyelles.
(Vous pouvez vous servir de la question précédente même si vous ne l'avez pas traitée.)

Exercice 7 *3 points*

Coder une fonction **extremes** qui prend en paramètre une liste **lst** et qui renvoie la différence entre le dernier élément de **lst** et le premier élément de **lst**.

Si la liste `lst` est vide, la fonction doit afficher le message "liste vide !" et renvoyer `None`.

Exemple d'utilisation :

```
>>> extremes([2, 5, 3, 6])
4
>>> extremes([])
'liste vide !'
```

Code de votre fonction :