

# DS 2

Informatique pour tous, première année

Julien REICHERT

**Exercice 1 :** Écrire une fonction `produit_f` qui a pour arguments une fonction `f` et un entier `n` et qui renvoie le produit  $f(0)f(1)\dots f(n)$ . Écrire ensuite une fonction `fact` qui utilise `produit_f` et qui renvoie la factorielle de son argument.

Indication : Pour la deuxième partie, on peut commencer par définir la fonction qui sera en argument de `produit_f`.

**Exercice 2 :** Écrire une fonction qui parcourt une liste en argument et qui détermine s'il n'existe pas deux éléments égaux dans cette liste.

Pour les exercices suivants, on rappelle la notion de *slice* (« tranche ») : Soit `seq` une séquence (liste ou chaîne de caractères notamment), alors pour `i`, `j` et `n` trois entiers (`n` non nul), `seq[i:j:n]` (ou, de manière équivalente, `seq[slice(i, j, n)]`) est l'objet de même type que `seq` contenant les éléments de `seq` entre les indices `i` (inclus) et `j` (exclu) par pas de `n` (l'objet sera vide si les indices progressent dans le mauvais sens).

Si le deuxième double-point n'existe pas, on considère que `n` vaut 1. En pratique, ne rien mettre d'un côté d'un double-point donne à un entier sa valeur la plus générale possible (ainsi, `seq[2:]` regroupe par exemple tous les éléments de `seq` à partir du troisième et jusqu'à la fin).

Il est important de souligner que si `i` ou `j` débordent de l'intervalle autorisé, la tranche s'arrête à la fin de la séquence sans déclencher d'erreur. De plus, si `i` est négatif (idem pour `j`), il est remplacé par l'indice positif correspondant (par exemple `-1` devient `len(seq)-1`).

**Exercice 3 :** Déterminer en la justifiant la valeur des variables `seq1`, `seq2`, `seq3` et `seq4` dans le programme suivant.

```
l = []
for i in range(100):
    l.append(i*i)

s = "QUEL EST LE PRODUIT DE SIX PAR NEUF"

seq1 = s[3:6]
seq2 = l[50:80:5]
seq3 = s[4:-5]
seq4 = l[-1:1:-25] + l[1:1000:100]
```

**Exercice 4 :** Écrire une fonction ayant pour arguments une liste `seq` et trois entiers `i`, `j` et `n` et retournant `seq[i:j:n]`. Pour cette fonction, il est interdit d'utiliser la notation présentée ci-dessus, ainsi que la fonction `slice`.