

1 Usages élémentaires de la boucle for

Somme : $\sum_{k=0}^n a_k$	s=0 for k in range(0,n+1): s=s+ak
Produit $\prod_{k=0}^n a_k$	p=1 for k in range(0,n+1): p=p*ak
Conjonction de variables binaires : $(a_0 \text{ and } a_1 \text{ and } \dots \text{ and } a_n)$	v=True for k in range(0,n+1): v=v and ak
Disjonction de variables binaires : $(a_0 \text{ or } a_1 \text{ or } \dots \text{ or } a_n)$	v=False for k in range(0,n+1): v=v or ak
Suite récurrente : $\begin{cases} u_0 = u_0 \\ u_{n+1} = f(u_n) \end{cases}$	u=u0 for k in range(0,n+1): f=f(u)
Suite récurrente d'ordre 2 : $\begin{cases} u_0 = u_0 \\ u_1 = u_1 \\ u_{n+2} = f(u_n, u_{n+1}) \end{cases}$	u,v=u0,u1 for k in range(0,n+2): w=v v=f(u,v) u=v
Parcours d'une liste L=[a0, ..., an] :	for i in range(len(L)): print(L[i]) ou for x in L: print(x)
Construction d'une liste [a0, ..., an] où ak=f(k)	L=[] for k in range(0,n+1): L.append(f(k)) ou L=[f(k) for k in range(0,n+1)]

2 Comparaison des listes & arrays

	list	array
type	natif	importé du module numpy
contenu	hétérogène	homogène
opérations	« + » concaténation « * » répétition	« + » addition terme à terme « * » multiplication terme à terme