Étude de permutations

L'objet du TP est d'étudier la représentation des permutations d'un ensemble fini et les opérations usuelles sur ces permutations.

En Maple, on utilisera comme ensemble fini à n éléments l'intervalle entier $[\![1, n]\!]$ et on représentera une permutation σ de $[\![1, n]\!]$ par un tableau t tel que $t[i] = \sigma(i)$ pour $i = 1, \dots, n$. La longueur d'un tableau t sera donnée par la fonction (à saisir) : `longueur := a -> nops([entries(a)]);`

En Caml, on utilisera comme ensemble fini à n éléments l'intervalle entier $[\![0, n-1]\!]$ et on représentera une permutation σ de $[\![0, n-1]\!]$ par un tableau t tel que $t.(i) = \sigma(i)$ pour $i = 0, \dots, n-1$.

1) Vérification

Écrire une fonction `verif` qui prend un tableau d'entiers en argument et retourne un booléen disant si ce tableau représente bien une permutation.

2) Orbite d'un élément

Écrire une fonction `orbite` prenant en arguments un tableau t supposé représenter une permutation σ et un entier i dans l'intervalle de permutation ($[\![1, n]\!]$ en Maple et $[\![0, n-1]\!]$ en Caml) et retournant l'orbite de i selon σ, représentée par un tableau contenant les différents éléments $\sigma^k(i)$ dans l'ordre.

3) Décomposition en cycles

A l'aide de la fonction `orbite` précédente, écrire une fonction `cycles` prenant en argument un tableau t représentant une permutation σ et retournant la décomposition de σ en cycles à supports disjoints, sous forme d'un tableau c dont chaque élément est l'un des cycles de la décomposition.

4) Applications

Utiliser la fonction `cycles` pour déterminer :
a) la signature d'une permutation ;
b) l'ordre d'une permutation ;

5) Le jeu de cartes

On dispose d'un jeu de 2^n cartes numérotées de 0 à $2^n - 1$ initialement rangées par ordre croissant (la carte numéro 0 est en bas du tas) et on bat le jeu de la manière suivante :

– découper le paquet en deux paquets de 2^{n-1} cartes ;
– reformer un seul tas en retournant alternativement une carte de chaque paquet et en la plaçant sur les cartes déjà retirées, façe vers le haut.
– retourner le tas pour que les cartes aient leur face vers le bas.

Soit σ_n la permutation de $[\![0, 2^n - 1]\!]$ ainsi obtenue. Programmer le calcul de σ_n et étudier expérimentalement l'ordre de σ_n en fonction de n. Donner une interprétation de σ_n en termes de numération binaire.