

# algorithme de Huffman

**Données:** un alphabet  $\mathcal{L}$ ,  
avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$   
**si**                      **alors**

**sinon**

fin

**Données:** un alphabet  $\mathcal{L}$ ,  
avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$   
**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$   
**si**  $\mathcal{L}$  *a deux lettres* **alors**  
  
**sinon**

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,  
avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$   
**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$   
**si**  $\mathcal{L}$  a deux lettres **alors**  
    encoder une des lettres avec 0 et l'autre avec 1;  
**sinon**

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

Construire récursivement un code préfixe  $\delta$  optimal pour  $\mathcal{M}$

d'arbre de code  $U$ ;

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

Construire récursivement un code préfixe  $\delta$  optimal pour  $\mathcal{M}$

d'arbre de code  $U$ ;

Définir l'arbre de code préfixe pour  $\mathcal{L}$

**fin**



**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

Construire récursivement un code préfixe  $\delta$  optimal pour  $\mathcal{M}$

d'arbre de code  $U$ ;

Définir l'arbre de code préfixe pour  $\mathcal{L}$  en remplaçant la feuille de

$U$  correspondant à la lettre

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

Construire récursivement un code préfixe  $\delta$  optimal pour  $\mathcal{M}$

d'arbre de code  $U$ ;

Définir l'arbre de code préfixe pour  $\mathcal{L}$  en remplaçant la feuille de

$U$  correspondant à la lettre  $l_{12}$  par un neud interne d'où partent

deux feuilles correspondant à

**fin**

**Données:** un alphabet  $\mathcal{L}$ ,

avec pour chaque lettre  $l_i$  une fréquence  $f_{l_i}$

**Résultat:** un arbre de codage préfixe optimal de  $\mathcal{L}$

**si**  $\mathcal{L}$  a deux lettres **alors**

encoder une des lettres avec 0 et l'autre avec 1;

**sinon**

soient  $l_1$  et  $l_2$  les deux lettres de plus basses fréquences;

fabriquer un nouvel alphabet  $\mathcal{M}$  en supprimant de  $\mathcal{L}$   $l_1$  et  $l_2$  et

en les remplaçant par la lettre  $l_{12}$  de fréquence  $f_{l_1} + f_{l_2}$ ;

Construire récursivement un code préfixe  $\delta$  optimal pour  $\mathcal{M}$

d'arbre de code  $U$ ;

Définir l'arbre de code préfixe pour  $\mathcal{L}$  en remplaçant la feuille de

$U$  correspondant à la lettre  $l_{12}$  par un neud interne d'où partent

deux feuilles correspondant à  $l_1$  et  $l_2$ ;

**fin**