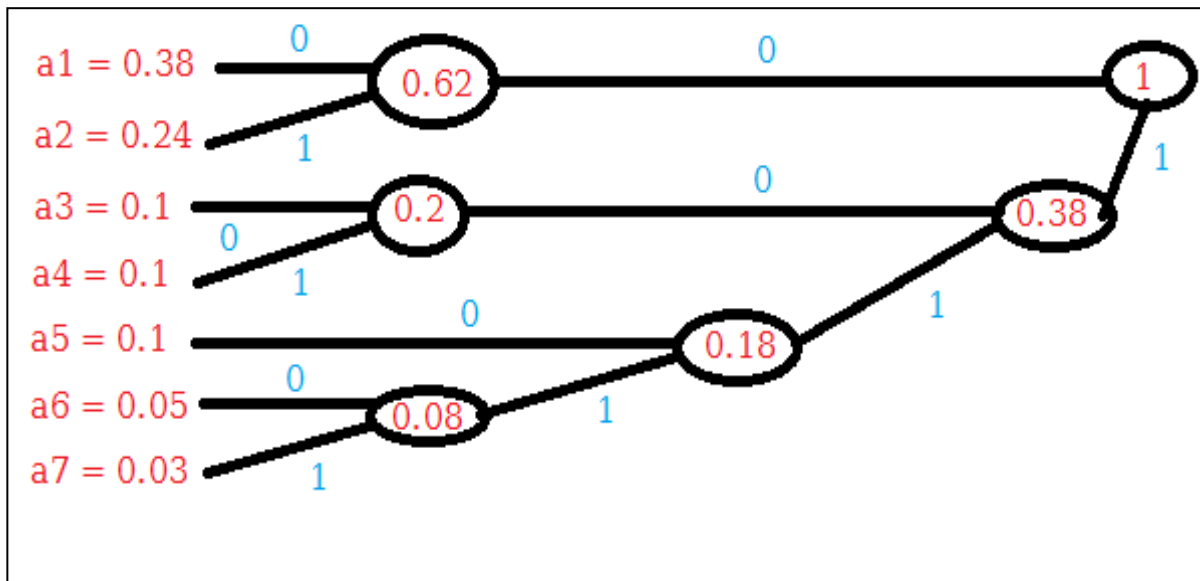


Module :
TP Codage et compression

TP N° : 01

Codage de Huffman



Objectifs du TP :

- Effectuer le codage et le décodage d'une chaîne de symboles numériques en utilisant l'algorithme de Huffman.
- Exploiter quelques fonctions de Matlab.

Manipulation :

Considérons le message de source suivant :

```
symbols = {'C', 'A', 'B', 'F', 'G', 'E', 'D', 'H'};  
p = [0.3 0.2 0.18 0.1 0.07 0.06 0.05 0.04];
```

1. Calculer l'entropie H de ce message ?
2. Trouver analytiquement le dictionnaire de Huffman adéquat pour ces symboles ?
3. Calculer la longueur moyenne L de code de cette source ?
4. Maintenant en utilisant les fonctions Matlab (**huffmandict**), trouver le dictionnaire de ces symboles.
6. Comparer les deux codes (que tu as obtenu analytiquement avec ceux calculés par Matlab) ?

Manipulation 2 :

Dans cette manipulation et toujours sur Matlab, on garde le même dictionnaire obtenu dans la première manipulation mais cette fois-ci on essaie de coder un nouveau message s'écrit comme suit :

msg = { FACHEGAD }

1. Trouver le code du Huffman qui compresse **msg** en utilisant la fonction **huffmanenco**.
2. Utiliser la fonction Matlab **huffmandeco** pour décompresser la séquence binaire fournie par le compresseur d'Huffman.
3. Comparer la taille initiale de **msg** (1 octet pour chaque symbole) avec la taille du code fourni par le compresseur d'Huffman.
4. Conclure ?
5. Calculer sur Matlab le taux de compression. Que vous remarquez.

Remarque : $\text{taux_compression} = (1 - (\text{taille de huffman} / \text{taille_originale})) * 100$