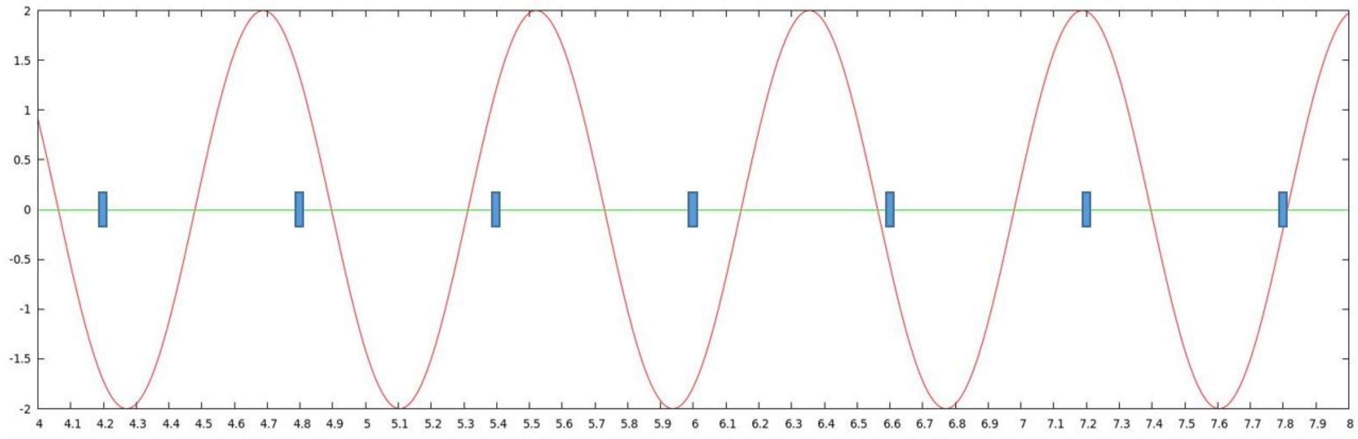
Solution QUIZZ et questions ouvertes MT-EL 25/11/2016

Remarque : l'ordre des réponses était différent selon les variantes. Donc ne faites pas attention à la lettre correspondant à la réponse correcte mais seulement à la réponse correcte elle-même qui est surlignée en jaune. Quelques explications brèves sont ajoutées pour pouvoir retrouver la bonne réponse des questions plus difficiles.



Question 1 : la sinusoïde X(t) apparaissant dans le dessin ci-dessus est échantillonnée aux instants indiqués par une petite barre verticale sur l'axe temporel (unité en secondes). On reconstruit le signal à l'aide de la formule d'interpolation.

Indiquer dans quel intervalle (en Hz) se situe la fréquence f_I de la sinusoïde reconstruite :

A f_I est supérieure ou égale à 1.4
B 1. $< f_I < 1.4$
C 0.55 $< f_I < 1.$
D 0.3 $< f_I < 0.55$ // d'abord mesurer les périodes sur le dessin puis raisonner sur les fréquences

---

Soit le signal $X(t) = 2\sin(8\pi t) + \sin(32\pi t) + \pi$

Question 2 : la bande passante de X(t) est (en Hz)
A 8
B 32
C 16
D 16π

Question 3 : on filtre X(t) avec un filtre à moyenne mobile de période d'intégration 0,25s. Le signal filtré est :
A $2\sin(2\pi t) + \sin(8\pi t) + \pi/4$
B π // la période d'intégration est un multiple entier de la période des 2 sin() on obtient donc zéro pour elles mais il reste le terme constant π
C $2\sin((\pi/4)t)$
D Aucune des autres réponses

Question 4 : on filtre X(t) avec un filtre Passe-haut de fréquence de coupure 30 Hz. La bande passante du signal filtré est :

A identique

B réduite à zéro // car la bande passante de X(t) est 16Hz.
Il ne reste aucune fréquence au dessus de 30 Hz

C doublée

D plus faible mais pas réduite à zéro

Question 5 : On vous donne le dictionnaire suivant :

A	B	C	D	E
00	01	10	110	111

Indiquer la proposition qui code le mot DECADE :

A 1101111101110111

B 1101111000110111 // on décode de la gauche vers la droite

C 1101101010110111

D 1101101011010111

Un message comporte les nombres d'apparitions suivants pour les lettres de A à F :

Lettre	A	B	C	D	E	F
Nb apparitions	18	9	8	7	7	2

Question 6 : soit un codage de **Shannon-Fano** pour le tableau ci-dessus. Indiquer la proposition vraie :

A Les codes de toutes les lettres ont la même longueur

B Les lettres A et B ont un code de 2 bits, tandis que C,D,E et F ont un code de 3 bits
// car le découpage est plus équilibré avec les groupes (A,B) et (C,D,E,F)

C Les lettres A,B et C ont un code de 2 bits, tandis que D,E et F ont un code de 3 bits

D Le nombre moyen de bits par lettre est égal à l'entropie du message

Question 7 : pour le codage de **Shannon-Fano**, combien de bits au total faut-il pour coder le message suivant : DEADBEEF

A 19

B 24

C 22 // avec la table fournie ci-dessus (les 3 questions sont groupées)

D 23

Question 8 : Indiquer la performance de ce codage de **Shannon-Fano** en nombre moyen de bit par lettre:

A 2.47 // avec la table fournie ci-dessus (les 3 questions sont groupées)

B 3.12

C 2.32

D 2.75

Question 9 : Soit les deux signaux suivants :

$$X_1(t) = \sin(30\pi t)$$
$$X_2(t) = \sin(24\pi t + 6\pi)$$

Quelle est la période du signal $X_1(t) + X_2(t)$?

A 1/54
B 1/12
C 1/15
D 1/3 // car 1/3 = 4*(1/12) = 5*(1/15)

Question 10 : Lequel des signaux suivants ne peut PAS être parfaitement reconstruit après un échantillonnage avec une fréquence de 4.1 Hz

A $X_1(t) = \cos(-3\pi t) + \cos(3\pi t)$
B $X_2(t) = \cos(2\pi t) \sin(3\pi t - \pi/3)$ // utiliser la formule qui transforme en une somme
C $X_3(t) = \sin(2\pi t + \pi/3) + \sin(3\pi t - \pi/3) + \sin(4\pi t + \pi/3)$
D $X_4(t) = \sin(5\pi t) + \sin(5\pi t - \pi)$ // c'est la fonction nulle , pas la bonne réponse. aucun problème pour la reconstruire

----

Question 11 : Lequel de ces mots possède la plus grande entropie

A MASSACHUSETTS
B MACHAUSSETTE // au besoin, utiliser le graphe de la fonction log2
C MISSISSIPPI
D MISSOURI

Question 12 : la valeur de la plus faible entropie est environ

A 1.50
B 1.82 // MISSISSIPPI au besoin, utiliser le graphe de la fonction log2
C 2.08
D 1.32

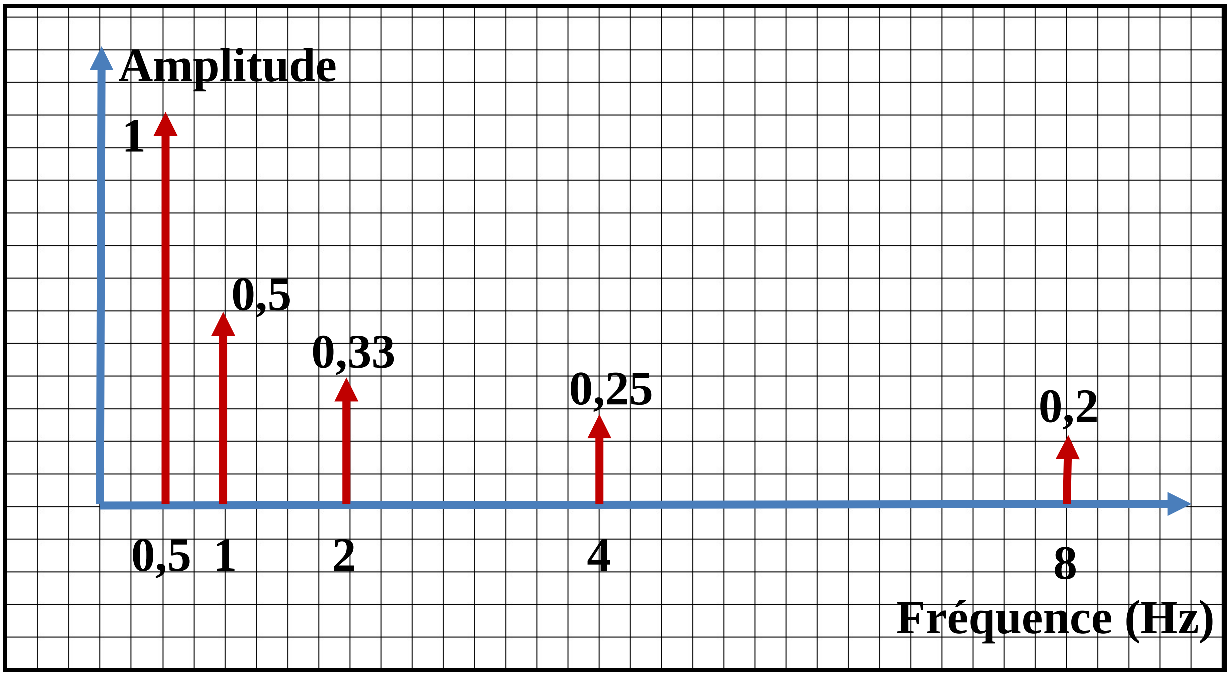
Questions Ouvertes

Question 1 : Echantillonnage

Soit le signal :

$$X(t) = \sum_{i=0}^{4} \frac{\sin(2^i \pi t)}{i+1}$$

a) dessiner son spectre de fréquence :



b) Un dispositif permet l'échantillonnage de X(t) mais avec seulement une fréquence d'échantillonnage maximum de 9Hz. Comment procédez-vous pour garantir que le signal interpolé $X_I(t)$ à partir des échantillons ne présente pas d'effet stroboscopique ? Fournissez des éléments quantitatifs pour justifier votre réponse :

Ce signal présente une bande passante de 8Hz. Or la fréquence d'échantillonnage est de 9Hz et si on l'utilise sur le signal brut cela va créer un effet stroboscopique.

Il faut donc filtrer la fréquence maximum de 8Hz à l'aide d'un filtre passe-bas idéal.

La fréquence de coupure de ce filtre doit conserver toutes les autres fréquences de X(t) car la plus importante ensuite est seulement de 4Hz qui ne va pas créer d'effet stroboscopique car elle est inférieure à fe/2 = 4.5Hz.

En résumé, la fréquence de coupure fc du filtre passe-bas doit être telle que 4Hz < fc < 8Hz

Le signal ainsi filtré pourra être reconstruit parfaitement à partir des échantillons obtenus avec fe = 9Hz

Question 2 : Codage de Huffman

Soit un alphabet contenant ***n*** symboles et un message **X** dans lequel chaque symbole s_i apparait a_i fois tel que : $a_i = 2^{i-1}$. Ainsi, le symbole s_1 apparait $2^{1-1} = 2^0$ fois, c'est-à-dire une seule fois, tandis que le symbole s_n apparait 2^{n-1} fois dans le message **X**.

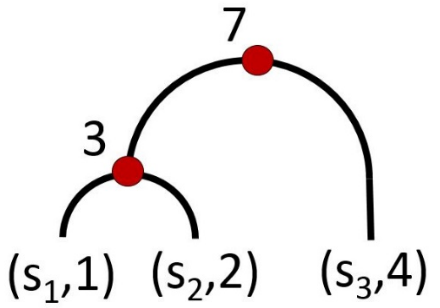
A partir de cette définition nous pouvons établir la longueur totale ***m*** du message **X** avec cette formule :

$$m = \sum_{i=1}^{n} 2^{i-1} = 1 + 2 + 4 + \cdots + \ 2^{n-1}.$$

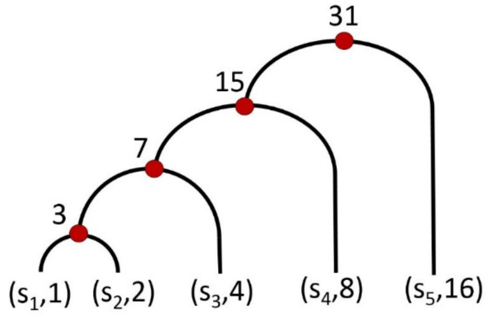
a) Dessiner l'arbre qui permet de construire le **code de Huffman** pour les cas suivants :

le dessin doit indiquer la taille des regroupements des nœuds internes pour avoir tous les points

n = 3



n = 5



b) Maintenant, dans le cas général pour ***n***, justifier combien de bits sont nécessaires pour construire le code du **symbole le plus fréquent**.

Pour justifier cette réponse (<u>et avoir tous les points</u>) nous avons besoin de rappeler l'égalité suivante : **la somme des 2^i, pour i de 0 à N-1, = 2^N -1**

Nous avons rencontré cette égalité dans le chapitre4 du module1 pour construire le complément à deux d'un entier sur N bit. Cette équation peut aussi être établie à partir de la formule qui donne la somme des termes d'une suite géométrique, appliquée dans notre cas pour **m**, ce qui donne **m = $1*(1+\ldots+2^{N-1}) = (1-2^N)/(1-2) = 2^N - 1$**

Ou encore : **$2^N = 2*2^{N-1} = 2^{N-1} + 2^{N-1} = 2^{N-1} + 2*2^{N-2} = 2^{N-1} + 2^{N-2} + 2^{N-2} = 2^{N-1} + 2^{N-2} + \ldots + 2^0 + 2^0$**

D'où, en faisant passer un 2^0 à gauche du signe égal : **$2^N - 1 = 2^{N-1} + 2^{N-2} + \ldots + 2^0$**

Cette égalité permet de prouver que la méthode de Huffman va procéder par regroupement systématique en commençant par les 2 symboles s1 et s2 associés aux puissances 0 et 1 dont la somme est plus petite <u>d'une unité</u> que le nombre d'apparitions du symbole s_3 (équation pour N valant 2). Ensuite, cette même équation établit que chaque regroupement des puissances de 2, de 0 à K-1, est plus petit <u>d'une unité</u> que la puissance $\mathbf{2^K}$. Et seul ce $\mathbf{2^K}$ permet de construire le prochain regroupement le plus petit. C'est donc la puissance de 2 la plus grande qui sera regroupée en dernier au groupe constitué de la somme de toutes les autres puissances de 2 qui lui sont inférieures. Du fait de ce dernier regroupement, le symbole le plus fréquent associé à cette plus grande puissance de deux (2^{n-1}) n'aura besoin que de ***un seul bit*** pour être codée.

c) Toujours dans le cas général pour ***n***, justifier combien de bits sont nécessaires pour construire le code du **symbole le moins fréquent**.

L'arbre construit pour n symboles a une profondeur de (n-1) car les 2 symboles les moins fréquents sont d'abord regroupés (= 1bit) puis chaque symbole supplémentaire parmi les (n-2) restant ajoute un bit supplémentaire du fait de la propriété mentionnée ci-dessus (la somme des puissances de 2 de 0 à N-1 est inférieure de une unité à 2^N). Nous avons donc le bilan suivant : <u>nombre de bits = profondeur des nœuds internes de l'arbre</u> = 1 + (n-2) = ***n-1 bits***