

<p>- 1_SPE NSI_4 DS02 Introduction à la programmation Binaire - Circuit - Codage Algorithme de tri</p>	<p>Le 14/12/2021 Durée: 1h55</p>
<p>Calculatrice non autorisée Aucun document autorisé Réponses sur feuilles doubles numérotées</p>	

<https://www.mathexien.com>

Exercice 1: (10 points)

<p>1.1. Coder le nombre décimal $(9)_{10}$ en binaire</p>	<p>1.2. Décoder le nombre binaire $(1011)_2$ en décimal</p>
<p>1.3. Coder le nombre décimal $(99)_{10}$ en binaire</p>	<p>1.4. Décoder le nombre hexadécimal $(3B)_{16}$ en décimal</p>
<p>1.5. Ecrire ce que va afficher la programme ci-dessous:</p> <pre> a = [1 , 2 , 3 , 4] b = [a , a , 1 , 2] print(a[1]) print(b[1][3]) </pre>	
<p>1.6. Implémenter une fonction qui:</p> <ol style="list-style-type: none"> 1. Demande à l'utilisateur: 'Avez vous bien révisé? (Répondre Oui ou Non)' 2. Récupère la réponse de l'utilisateur 3. Revient à l'étape 1. si la réponse est différente de 'Oui' et de 'Non' 4. Retourne la valeur 1 si la réponse est 'Oui' et la valeur 0 si la réponse est 'Non' 	
<p>1.7. Implémenter une fonction qui renvoie le maximum d'une liste de nombres donnée en paramètre.</p>	
<p>1.8. Recopier et compléter le code ci-dessous (remplacer les XXX):</p> <pre> 1 def conversion_decimal_binaire(n): 2 binaire = "" 3 dividende = n 4 while XXX != 0: 5 quotient = dividende // 2 6 binaire = binaire + str(XXX) 7 dividende = XXX 8 binaire_inv = "" 9 for i in range(len(binaire)): 10 binaire_inv = binaire_inv + binaire[XXX] 11 return XXX </pre>	

Licence CC BY-NC-SA 4.0

Exercice 2: (5 points) <https://www.mathexien.com>

Implémenter une fonction qui trie une liste de nombres.

Exercice 3 (5 points)

Un multiplexeur est un circuit qui permet de choisir une entrée parmi plusieurs en fonction de la valeur d'une commande.

Pour un multiplexeur 2 entrées (e_1 et e_0), la commande sur 1 bit (c_0) permet de sélectionner pour sortie (S) une des deux entrées:

- si $c_0 = 0$ alors $S = e_0$
- si $c_0 = 1$ alors $S = e_1$

Ainsi sa table de vérité se présente comme suit:

c_0	e_1	e_0	S
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Pour un multiplexeur 4 entrées ($e_3 e_2 e_1 e_0$), la commande sur 2 bits ($c_1 c_0$) permet de sélectionner pour sortie (S) une des quatre entrées:

- si $c_1 c_0 = 00$ alors $S = e_0$
- si $c_1 c_0 = 01$ alors $S = e_1$
- si $c_1 c_0 = 10$ alors $S = e_2$
- si $c_1 c_0 = 11$ alors $S = e_3$

3.1. Déterminer la fonction logique non simplifiée de S pour un multiplexeur 2 entrées

3.2. Déterminer une fonction logique simplifiée de S pour un multiplexeur 2 entrées

3.3. Représenter un circuit simplifié d'un multiplexeur 2 entrées

3.4. Déterminer une fonction logique simplifiée de S pour un multiplexeur 4 entrées

3.5. Représenter un circuit simplifié d'un multiplexeur 4 entrées

Licence CC BY-NC-SA 4.0

En option (++): expliquer comment utiliser un multiplexeur pour récupérer sur une mémoire d'un kilo octets des informations codées sur un octet