

- 1_SPE NSI5	Le 16/11/2020 Durée: 1h30
DS01 Thème 1: Bases de la programmation Thème 2: Binaire – Circuit – Codage	
Calculatrice non autorisée Aucun document autorisé Réponses sur feuilles doubles numérotées	

<http://www.mathexien.com>

**Exercice 1: (10 points)**

- 1.1. Donner l'écriture binaire du nombre dont l'écriture décimale est 5
- 1.2. Donner l'écriture hexadécimale du nombre dont l'écriture binaire est 1 0101
- 1.3. Donner l'écriture décimale du nombre dont l'écriture hexadécimale est 20
- 1.4. Etablir la table de vérité de la fonction:  $(x \text{ And } \text{Not}(y)) \text{ Or } (\text{Not}(x) \text{ And } y)$
- 1.5. Programmer une instruction qui affiche le carré d'un nombre entier entré par l'utilisateur depuis le clavier
- 1.6. Donner la valeur des variables après exécution du programme ci-dessous:

```
a = 3
b = a
a = a + 1
b = b * 2
c = b % a
```

- 1.7. Ecrire ce qui sera affiché après exécution du programme ci-dessous:

```
a = 7
if a % 2 == 0 :
    print(a)
elif a / 2 != 0 :
    print("OK")
else :
    print("KO")
```

- 1.8. Décrire le traitement réalisé par la fonction ci-dessous ( a et b sont des nombres entiers):

```
def maFonction( a , b ):
    c = 2
    for i in range( c ):
        a = a * a
    return a
```

- 1.9. Décrire ce qui se passera après appel de la fonction ci-dessous (a est un nombre entier):

```
def maFonction( a ):
    i = 10
    while i < a :
        print("NSI")
```

- 1.10. Donner la valeur des variables c et d après exécution du programme ci-dessous:

```
a = [ 1 , 2 , 3 , 4 , 5 ]
b = [ a , a , a ]
c = b [ 3 ]
d = b [ 1 ] [ 0 ]
```

## Exercice 2: (7 points)

<http://www.mathexien.com>

Programmer une fonction qui:

- prend une liste de nombres entiers en paramètre
- renvoie une liste contenant le minimum et le maximum des nombres de la liste donnée en paramètre

$a = \text{minMax}([1, 2, 3, 4])$                        $\# a = [1, 4]$

$b = \text{minMax}([7, -3, 2, -5, 9, 1, 6])$                        $\# b = [-5, 9]$

Vous ne pouvez pas utiliser de fonction native.

## Exercice 3: (7 points)

Un comparateur permet de comparer deux nombres en indiquant en sortie le résultat de la comparaison.

On étudie un comparateur deux bits qui a:

- en entrée deux nombres  $A$  et  $B$  codés en binaire sur deux bits:
  - $A : A_1A_0$
  - $B : B_1B_0$
- en sortie trois indicateurs informant sur la comparaison correspondante ( 1 si VRAI et 0 sinon):
  - $S$  (pour supérieur): 1 si  $A > B$  et 0 sinon
  - $I$  (pour inférieur): 1 si  $A < B$  et 0 sinon
  - $E$  (pour égal): 1 si  $A = B$  et 0 sinon

Etablir la table de vérité de ce comparateur.

## Exercice 4: (6 points)

Vous réalisez en binôme un projet sous Python avec le module Turtle où vous devez dessiner un pavage du plan d'un motif obtenu par rotation d'une forme de base.

Vous devez développer la fonction motif ( taille , n ) et votre binôme l'utilisera pour réaliser le pavage.

La fonction motif ( taille , n ):

- utilise la fonction formeDeBase( taille ) qui:
  - prend en paramètre taille: une taille de la forme de base (int)
  - dessine une forme de base de votre choix
- prend en paramètres:
  - taille: une taille de la forme de base (int)
  - n: le nombre de formes de base dans le motif (int)
- dessine un motif obtenu par n rotations de la forme de base

Pour turtle, on utilisera uniquement les fonctions:

- forward ( x )     $\#$  avance de x pixels
- right ( y )       $\#$  tourne de y degrés vers la droite
- penup ( )         $\#$  pour lever le stylo
- pendown ( )      $\#$  pour descendre le stylo

Programmer les fonctions formeDeBase( taille ) et motif ( taille , n )