

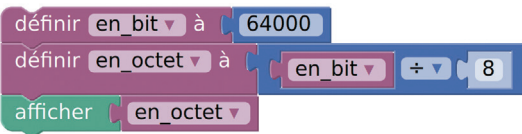
Au collège, vous avez découvert la programmation par blocs avec Scratch. Nous allons maintenant programmer avec le langage Python, un langage simple d'usage et largement répandu, utilisé au lycée.

Aller sur le site web lycee.editions-bordas.fr/cahier-SNT, séquence 15, et choisir l'interface « Du langage par blocs à Python ».

Cette interface permet d'écrire un programme avec des blocs et d'obtenir en temps réel sa traduction en langage Python.

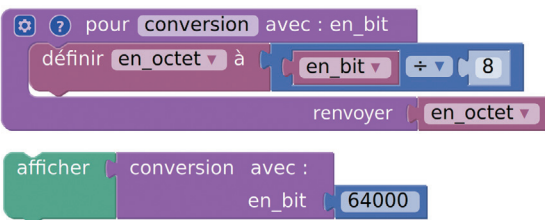
1. Le programme ci-dessous, écrit en langage par blocs, convertit un débit binaire exprimé en bit par seconde (bit/s) en octet par seconde (o/s) (un octet est composé de 8 bits).

Saisir le programme avec le langage par blocs, puis recopier le programme Python associé.



Le programme précédent se contente d'afficher la valeur calculée, ce qui n'est pas pratique pour effectuer d'autres traitements sur celle-ci. Nous allons donc le modifier.

2. a. Saisir le programme ci-dessous, écrit en langage par blocs, à l'aide du bouton [réglage]. Puis tester le programme en cliquant sur [exécution].



b. Recopier les instructions en langage Python qui définissent la fonction `conversion()`, en faisant attention à l'alignement en début de ligne.

En langage Python, le mot-clé `def` permet de définir une **fonction**. La ou les valeurs passées en entrée de la fonction sont appelées **argument(s)**. Dans le corps de la fonction, les instructions sont **indentées**, c'est-à-dire décalées vers la droite. Le mot-clé `return` permet de renvoyer la ou les valeur(s) calculé(s) par la fonction.

c. Dans la fonction `conversion()` écrite en langage Python, souligner l'argument et entourer la valeur renvoyée par celle-ci.

3. Dans votre calculatrice ou dans l'interface Python disponible sur lycee.editions-bordas.fr/cahier-SNT (séquence 15), saisir la fonction Python `conversion()` de la question 2.b.

4. Dans la console, exécuter l'instruction `conversion(64000)` et interpréter le résultat affiché.

On souhaite à présent exprimer le débit en kilooctet par seconde (ko/s).

5. Sous la première fonction, saisir la fonction Python `conversion_octet_ko()` ci-dessous.

Puis exécuter `conversion_octet_ko(64000)` dans la console.

Noter le résultat affiché et donner son interprétation.

```
01 def conversion_octet_ko(en_octet):
02     en_ko = en_octet / 1000
03     return en_ko
```

Activité 2 / Les instructions conditionnelles



20 min

Un site d'impression de photos affiche les tarifs ci-contre. On souhaite programmer un algorithme qui a pour paramètre le nombre de photos à imprimer, et renvoie le prix de la commande.

Tarifs de nos tirages
De 1 à 50 photos : 0,18 €/photo
Plus de 50 photos : 0,17 €/photo

1. a. Accéder à l'interface « Du langage par blocs à Python » sur le site lycee.editions-bordas.fr/cahier-SNT. Saisir la fonction en blocs ci-dessous à l'aide du bouton , puis la tester.
- b. Recopier le programme correspondant écrit en langage Python.

```

pour calculer_prix avec : nombre_photos
si nombre_photos ≤ 50
faire définir prix à nombre_photos x 0.18
sinon définir prix à nombre_photos x 0.17
renvoyer prix
    
```

.....

.....

.....

.....

.....

.....

- c. Entourer les mots-clés du langage Python qui permettent d'exécuter des instructions en fonction d'une condition donnée.

Si la commande est supérieure à 100 photos, une promotion de 10 % s'applique sur le prix du tirage.

2. Modifier le programme en blocs afin de prendre en compte cette promotion. Recopier ci-contre les instructions supplémentaires en langage Python :

.....

.....

.....

Activité 3 / Boucles bornée et non bornée



30 min

Au cours de l'année 2018, 1 800 exaoctets de données ont été échangées sur Internet. En 2019, cette valeur est passée à 2 400 exaoctets, ce qui représente une augmentation d'environ 33 %.

1. Accéder à l'interface « Du langage par blocs à Python » sur le site lycee.editions-bordas.fr/cahier-SNT. Saisir les fonctions en blocs suivantes en utilisant le bouton et les tester.

```

pour estimer_annee avec : quantite_initiale
définir quantite_finale à quantite_initiale
répéter 5 fois
faire définir quantite_finale à quantite_finale x 1.33
renvoyer quantite_finale

afficher estimer_annee avec :
quantite_initiale 1800
    
```

```

pour estimer_annee_2 avec : quantite_initiale
définir annee à 2018
définir quantite_finale à quantite_initiale
répéter tant que quantite_finale ≤ quantite_initiale x 3
faire définir quantite_finale à quantite_finale x 1.33
définir annee à annee + 1
renvoyer annee

afficher estimer_annee_2 avec :
quantite_initiale 1800
    
```

2. À l'aide de l'affichage, relier chaque mot-clé du langage Python à son utilisation :
 - for • répéter les instructions indentées un certain nombre de fois.
 - while • répéter les instructions indentées tant que la condition est vraie.

On veut déterminer en quelle année la quantité de données échangées sur Internet aura triplé par rapport à 2018, en considérant que l'augmentation sera de 33 % tous les ans.

3. Lequel des deux programmes précédents permet de résoudre le problème ? Justifier.

.....

.....

Les exercices suivants sont réalisables sur calculatrice ou sur l'interface Python disponible sur la page web lycee.editions-bordas.fr/cahier-SNT, séquence 15.



25 min

EXERCICE 1 / Distance entre un récepteur GPS et un satellite

Un récepteur GPS a reçu, à 16 h 00 min 0 s 15 ms, le signal émis par un satellite à 15 h 59 min 59 s 910 ms. Le signal se propage à environ 300 000 km/s.

1. Écrire en langage Python une fonction `temps_en_secondes()` qui prend en argument un horaire donné en heure, minute, seconde et milliseconde, et qui renvoie son équivalent en seconde.

.....

.....

2. Écrire en langage Python une fonction `distance()` qui prend en arguments l'horaire d'arrivée du message et son horaire d'émission, en seconde, et qui renvoie la distance parcourue par le signal, en kilomètre.

.....

.....

3. Interpréter le résultat de l'instruction ci-contre.

```
>>> distance(temps_en_secondes(15, 59, 59, 910), temps_en_secondes(16, 0, 0, 15))
31499.99999877764
```

.....

EXERCICE 2 / Conversion de coordonnées GPS



20 min

Pour un devoir, Enzo a besoin de convertir des coordonnées GPS données en degré décimal (DD) en coordonnées données en degré, minute, seconde (DMS), et inversement. Pour exprimer une coordonnée GPS, l'unité de base est le degré d'angle (1 tour complet = 360°), puis la minute d'angle (1° = 60'), et la seconde d'angle (1' = 60"). Par exemple, une latitude de 12,345 °(DD) correspond à 12° 20' 42" (DMS).

1. Écrire une fonction Python `dms_a_decimale()` qui prend en argument les valeurs en degré, minute, seconde d'une coordonnée DMS, et qui renvoie la coordonnée GPS en degré décimal.

.....

.....

2. Écrire une fonction Python `decimale_a_dms()` qui prend en argument les coordonnées GPS (uniquement positives) en degré décimal, et renvoie les valeurs correspondantes en degré, minute, seconde.

.....

.....

.....

.....

3. Tester vos deux fonctions dans la console à l'aide des deux exemples ci-dessous.

```
>>> decimale_a_dms(12.345)
(12.0, 20.0, 42.0)
```

```
>>> dms_a_decimale(12, 20, 42)
12.345
```

EXERCICE 3 / Promotion sur les appareils photo numériques



20 min



Un site de vente en ligne propose une promotion sur les appareils photo numériques. Si l'appareil coûte moins de 100 €, la promotion est de 10 %. Si l'appareil coûte 100 € ou plus, la promotion est de 20 %. Les frais de port de 10 € sont offerts si le montant de la commande, après promotion (et hors frais de port), est supérieur à 150 €.

Écrire une fonction qui prend en argument le prix initial de l'appareil photo, et qui renvoie le prix final de la commande. La tester pour différentes valeurs.

.....

.....

.....

.....

.....

EXERCICE 4 / Trouver le bogue !



20 min



Lucas a lu que l'écran de son ordinateur affichait jusqu'à 16 millions de couleurs différentes. Pour savoir combien de bits sont nécessaires pour coder autant de couleurs, il a écrit la fonction Python ci-contre.

```
01 def compte_bits(valeur):
02     nb_bits = 1
03     while valeur < 2**nb_bits:
04         nb_bits = nb_bits + 1
05     return nb_bits
```

L'appel `compte_bits(16000000)` renvoie 1. Lucas se rend compte qu'il y a un bogue dans son programme.

1. Indiquer l'instruction correcte à écrire en troisième ligne de la fonction.

2. Combien de bits sont nécessaires pour coder 16 millions de couleurs ?

.....

.....

EXERCICE 5 / Une avalanche d'e-mails



20 min



Noëlle a reçu 10 000 e-mails cette année. Elle a remarqué que chaque année, elle reçoit 5 % d'e-mails supplémentaires par rapport à l'année précédente.

1. Que permet de calculer la fonction écrite en langage Python ci-dessous ?

```
01 def mails(nombre, t, n):
02     for i in range(n):
03         nombre = nombre * (1 + t / 100)
04     return nombre
```

2. Qu'écrire dans la console Python afin d'afficher le nombre d'e-mails que Noëlle recevra dans six ans ?

3. Modifier la fonction `mails()` afin qu'elle renvoie le nombre total d'e-mails reçus par Noëlle au cours de cette année et les n années suivantes.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....