Extrait du programme de 2^{de} :

- décrire des **algorithmes** en langage naturel ou dans un langage de programmation ;
- en réaliser quelques-uns à l'aide d'un **programme** simple écrit dans un langage de programmation textuel ;
- interpréter, compléter ou modifier des **algorithmes plus complexes**.

Un langage de programmation simple d'usage est nécessaire pour l'écriture des programmes informatiques. **Le langage choisi est Python**, langage interprété, concis, largement répandu et pouvant fonctionner dans une diversité d'environnements. Les élèves sont entraînés à **passer du langage naturel à Python et inversement**.

L'algorithmique a une place naturelle dans tous les champs des mathématiques et les problèmes ainsi traités doivent être en relation avec les autres parties du programme (fonctions, géométrie, statistiques et probabilité, logique) mais aussi avec les autres disciplines ou la vie courante.

À l'occasion de **l'écriture d'algorithmes** et de petits programmes, il convient de transmettre aux élèves l'exigence d'exactitude et de rigueur, et de les entraîner aux pratiques

• Utiliser les variables et les instructions élémentaires

Contenus

- **Variables informatiques** de type entier, booléen, flottant, chaîne de caractères.
- **Affectation** (notée ← en langage naturel).
- Séquence d'instructions.
- Instruction conditionnelle.
- Boucle bornée (**for**), boucle non bornée (**while**).

Capacités attendues

- Choisir ou déterminer le type d'une variable (entier, flottant ou chaîne de caractères).
- **Concevoir et écrire une instruction** d'affectation, une séquence d'instructions, une instruction conditionnelle.
- Écrire une formule permettant un calcul combinant des variables.
- **Programmer**, dans des cas simples, une boucle bornée, une boucle non bornée.
- Dans des **cas plus complexes** : lire, comprendre, modifier ou compléter un algorithme ou un programme.

Contenus

- Fonctions à un ou plusieurs arguments.
- Fonction renvoyant un nombre aléatoire. Série statistique obtenue par la répétition de l'appel d'une telle fonction.

Capacités attendues

- **Écrire** des fonctions simples ; lire, comprendre, modifier, compléter des fonctions plus complexes. Appeler une fonction.
- **Lire et comprendre** une fonction renvoyant une moyenne, un écart type. Aucune connaissance sur les listes n'est exigée.
- **Écrire des fonctions** renvoyant le résultat numérique d'une expérience aléatoire, d'une répétition d'expériences aléatoires indépendantes.

Extrait du programme de 1^{ère} spécialité :

L'enseignement de spécialité de mathématiques de classe de première vise la **consolidation des notions de variable, d'instruction conditionnelle et de boucle ainsi que l'utilisation des fonctions**. La seule **notion nouvelle est celle de liste** qui trouve naturellement sa place dans de nombreuses parties du programme et aide à la compréhension de notions mathématiques telles que les suites numériques, les tableaux de valeurs, les séries statistiques...

Comme en classe de seconde, les algorithmes peuvent être écrits en **langage naturel ou utiliser le langage Python**.

• Notion de liste

La **génération des listes en compréhension et en extension** est mise en lien avec la notion d'ensemble. Les conditions apparaissant dans les listes définies en compréhension permettent de travailler la logique. Afin d'éviter des confusions, on se limite aux listes sans présenter d'autres types de collections.

Capacités attendues

- **Générer une liste** (en extension, par ajouts successifs ou en compréhension).
- Manipuler des éléments d'une **liste** (ajouter, supprimer...) et leurs indices.
- Parcourir une **liste**.
- Itérer sur les éléments d'une **liste**.

Extrait du programme de 1^{ère} technologique (sauf série ST2A) :

La pratique de **l'algorithmique** et de la **programmation** se poursuit au cycle terminal. En continuité avec la classe de seconde, **le langage utilisé est Python**.

Le programme vise la consolidation des **notions de variable, d'instruction conditionnelle et de boucle** ainsi que l'utilisation des **fonctions**. La seule **notion nouvelle est celle de liste** qui trouve naturellement sa place dans de nombreuses parties du programme et aide à la compréhension de notions mathématiques telles que les suites numériques, les tableaux de valeurs, les séries statistiques...

Extrait du programme de T^{ale} spécialité :

page suivante

Extrait du programme de T^{ale} spécialité :

au point et exécuter un programme simple. Les classes de seconde et de première ont permis de consolider les acquis du collège (notion de variable, type, de variables, affectation, instruction conditionnelle, boucle notamment), d'introduire et d'utiliser la notion de fonction informatique et de liste. En **algorithmique et programmation**, le programme reprend les programmes de seconde et de première sans introduire de notion nouvelle, afin de consolider le travail des classes précédentes.

Les **algorithmes** peuvent être écrits **en langage naturel ou utiliser le langage Python**. On utilise le symbole « ← » pour désigner l'affectation dans un algorithme écrit en langage naturel. L'accent est mis sur la programmation modulaire qui permet de découper une tâche complexe en tâches plus simples.

L'**algorithmique** trouve naturellement sa place **dans toutes les parties du programme** et aide à la compréhension et à la construction des notions mathématiques.

Capacités attendues

- **Générer une liste** (en extension, par ajouts successifs ou en compréhension)
- **Manipuler** des éléments d'une **liste** (ajouter, supprimer...) et leurs indices.
- Parcourir une **liste**.
- Itérer sur les éléments d'une **liste**.

Extraits de sujets de bac :

Sujet "zéro" **2021** :

```
def seuil(p):  
    n = 1  
    while 1 - (5/6)**n <= p :  
        n = n+1  
    return n
```

Quelle est la valeur renvoyée par la commande **seuil(0,9)**? Interpréter cette valeur dans le contexte de l'exercice.

On considère le programme suivant écrit en langage Python :

```
def seuil() :  
    n = 0  
    r = 1  
    while abs(r-sqrt(2)) > 10**(-4) :  
        r = (2+r)/(1+r)  
        n = n+1  
    return n
```

Sujet prévu pour le 15 mars **2021**

(mais épreuve finalement annulée à cause de la crise sanitaire)

Sujet Amérique du Nord juin **2021**

(candidats libres seulement car autres épreuves annulées)

```
def menace()  
    u = 0,6  
    n = 0  
    while u > 0,02  
        u = 0,75*u*(1-0,15*u)  
        n = n+1  
    return n
```

Sujet Asie juin **2021**

(candidats libres seulement)

```
def suite(n) :  
    u = 1 000  
    for i in range(n) :  
        u = 0,9*u + 250  
    return u
```

Recopier et compléter le programme en Python ci-dessous de sorte que la valeur cherchée à la question précédente soit stockée dans la variable n à l'issue de l'exécution de ce dernier.

```
u = 10560  
n = 0  
while ..... :  
    u = .....  
    n = .....
```

Centres étrangers juin **2021**

candidats libres

Métropole

(session de rattrapage septembre **2020**)

Algorithme	
1.	$V \leftarrow 1$
2.	Pour i variant de 1 à n
3.	$U \leftarrow \frac{\dots(\dots+2)}{(\dots+1)^2}$
4.	$V \leftarrow \dots$
5.	Fin Pour

Bac juin **2019** métropole :

```
I ← 0  
Tant que 2I ≤ A  
    I ← I + 1  
Fin Tant que
```

etc.....

en fait les trois quarts des sujets de bac contiennent une question avec un algorithme, qui est le plus souvent écrit en langage python depuis 2021.

Aucun sujet d'annale ne traite pour le moment de "listes" car cette partie a été suspendue pour la session 2021, mais est de nouveau au programme en 2022.

Les listes sont un élément en algorithmique au programme de 1^{ère}.