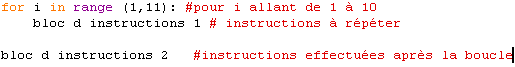
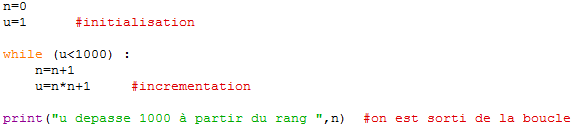
**TD5 : boucle for/boucle while**

**Syntaxe python de boucle for:**

****

**Syntaxe python de boucle while:**

****

**Exercice 0:**

1. Soit la suite définie par
2. que fait l'algorithme suivant?

Variables : n entier, u flottant

Entrée : un entier N

début

pour n allant de 0 à N faire:

u ← 795\*(0.9)\*\*n

Afficher(u)

fin Pour

fin

1. Ecrire un algorithme afin de trouver à partir de quelle valeur de n, devient inférieur à 300.
2. L’implémenter en Python
3. Soit la suite définie par
4. Ecrire un algorithme qui calcule et affiche
5. Ecrire un algorithme afin de trouver à partir de quelle valeur de n, devient supérieur à 9990.
6. L’implémenter en Python

**Exercice 1:**

1. Quel "jeu" l'algorithme suivant simule-t-il?

atrouver ← nombre aléatoire entre 0 et 50000

saisir essai

tant que (essai ≠ atrouver) répéter :

si (essai > atrouver) faire:

afficher ("c'est moins")

sinon :

afficher ("c'est plus")

saisir essai

fin de répéter

afficher ("gagner")

1. Lister les variables en précisant leur type
2. Recopier et compléter l'algorithme pour que l'on n'ait le droit qu'à 10 essais maximum.
3. Implémentez le en Python

**Exercice 2:**

On définit la suite par:

1. Ecrire et implémenter un algorithme qui calcule et affiche les 10 premiers termes de la suite.

Quel type de boucle choisissez-vous et pourquoi?

Etant donnée la question, où placez-vous l'affichage de la valeur de , dans ou après la boucle?

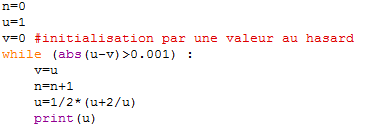
1. Cet algorithme, dit de Héron, permet d'obtenir une valeur approchée de .

Récrire votre algorithme afin que les termes soient calculés et affichés jusqu'à obtenir une précision (écart entre et ) inférieure à 0,001.

Quel type de boucle choisissez-vous et pourquoi ?

Vous devrez utiliser des fonctions mathématiques, aussi n'oubliez pas de saisir : from math import \* en début de programme. On rappelle que la racine carrée est donnée par sqrt() et que la valeur absolue est donnée par abs().

1. Que fait le programme suivant ? A quoi sert la variable v ?



1. L'algorithme de Héron permet d'approcher la racine carrée de tout nombre positif A comme limite de la suite définie par :

Modifier le programme précédent pour qu'il

* demande à l'utilisateur de choisir un nombre A positif
* calcule une valeur approchée de "à 0.001 près"
* affiche uniquement cette dernière valeur.

1. Pour aller plus loin, nous avons en quelque sorte "réécrit" la fonction sqrt contenue dans la bibliothèque math. pour achever son écriture en tant que fonction, il suffit de connaître la syntaxe de définition d'une fonction en Python :

