I Racine carré et entier

1.1 Description du problème

On se donne un entier naturel n et on souhaite savoir si $n = k^2$ pour un entier k, c'est à dire si $\sqrt{n}$ est encore entier.

Exercice 1
Trouver la plus petite puissance de 10 pour laquelle

```
int(math.sqrt(10**(2*p)))
```

ne retourne pas la valeur exacte.
Expliquer le phénomène.

1.2 A la recherche de la racine

Pour contourner le problème précédent, il faut travailler seulement avec des entiers. Le principe est d'utiliser la recherche par dichotomie sur les entiers entre 1 et n.

Exercice 2
Créer une fonction `racine_entiere(n)` qui retourne un couple (b, k) où b est un booléen et k un entier. b est la réponse à la question posée (est-ce que la racine carré de n est un entier ?) et k le plus grand entier tel que $k**2 <= n$ (la vraie partie entière de $\sqrt{n}$).

II Approximations

2.1 Suites récurrentes

Une bonne manière d'obtenir des valeurs approchées de certains nombres (pour nous $\sqrt{a}$ pour un nombre $a > 0$) et d'étudier des suites bien choisies de la forme

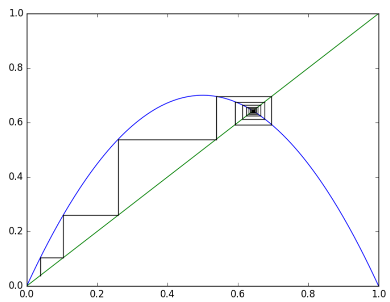
$$\begin{cases} u_0 \in \mathbb{R} \\ \forall n \in \mathbb{N}\ u_{n+1} = f(u_n) \end{cases}$$

où on a choisi le terme initial u_0 et la fonction f (d'une manière à préciser en cours de mathématiques).

Exercice 3
Créer une fonction `termes(f, u0, n)` qui retourne la liste des n premiers termes de la suite définie ci-dessus. f est une **fonction** python, $u0$ un nombre et n un entier naturel non nul.

Exercice 4
Tester la fonction précédente avec la fonction $f : x \mapsto \frac{x}{2} + \frac{1}{x}$, $u0 = 2$ et $n = 10$.

Exercice 5
On représente usuellement ce genre de suite par des graphiques qui ressemblent à ceci :



Créer une fonction `trace(f, a, b)` qui trace la courbe représentative de la fonction f (sur l'intervalle $[a, b]$) ainsi que la première bissectrice sur un même graphique.
Pour mémoire

```
import matplotlib.pyplot as plt
plt.plot(X, Y, color=''red'')
```

trace la courbe reliant les points dont les abscisses sont dans la liste X et les ordonnées dans la liste Y.

Exercice 6
Tester votre fonction avec `math.atan` comme fonction, sur $[-1, 1]$ puis avec une fonction que vous aurez créée.

Exercice 7
Créer une fonction `escargot(f, a, b, u0, n)` (sur le modèle de la fonction précédente) qui représente en plus la suite récurrente qui nous préoccupe.

2.2 Dichotomie

Exercice 8
Créer une fonction **dichotomie(a, epsilon)** qui recherche par dichotomie le point où la fonction $x \mapsto x^2 - a$ s'annule pour un $a > 0$. On arrête la recherche dès que la longueur de l'intervalle de recherche est inférieure ou égale à *epsilon*.
On pourra tenter de représenter graphiquement les points utilisés dans la recherche via la commande

```
plt.plot(X, Y, ''o'', color=''black'')
```

qui ne relie pas les points tracés.

III Bonus

Exercice 9
Pour a, b des entiers naturels non nul, calculer (sans flottant !) le quotient q de la division euclidienne de a par b. On pourra chercher q (entre 1 et a a priori) comme le plus grand entier tel que $a - bq \geq 0$. En déduire le reste de cette division.

Exercice 10
Imaginons qu'on ait trouvé une première approximation de $\sqrt{n}$ (comme à l'exercice 2). Notons r_1 cette approximation.
En cherchant r_2 sous la forme $r_2 = r_1 + a$ qui soit une meilleurs approximation de $\sqrt{n}$ (avec a "petit" devant r_1), donner un lien possible entre r_2 etr_1 et retrouver la formule de l'exercice 4.