
	2017/2018
 L1	Algorithmes et Programmation : Python 3

Pour chaque programme, rechercher les valeurs de test afin de parcourir toutes les branches et voir le comportement aux valeurs limites.

A0

Prise en main de l'environnement de programmation avec 'Hello World' ou la somme de deux entiers.

1. Utiliser python en mode ligne de commande
2. Executer un programme avec Idle
3. utiliser le transtypage.
4. Ecrire un programme de reconnaissance d'un nombre pair
5. Ecrire l algorithme de calcul de factoriel n! et le traduire en python. Prevoir les messages d'erreur à la saisies de n
- 5.En la réalité la bibliothèque math de python contient déjà une fonction $math.factorial(x)$ Return x factorial. Raises ValueError if x is not integral or is negative. La tester et l'utiliser la pour calculer et afficher le nombre d'euler.

A1 Ecrire et tester un programme de resolution d'une équation du second degré.

A2 Ce jeu est très simple. L'ordinateur tire un nombre au hasard entre 1 et 30 et vous avez cinq essais pour le trouver. Après chaque tentative, l'ordinateur vous dira si le nombre que vous avez proposé est trop grand, trop petit, ou si vous avez trouvé le bon nombre.

Exemple de partie

```
J'ai choisi un nombre entre 1 et 30
A vous de le deviner en 5 tentatives au maximum !
Essai no 1
Votre proposition : 15
Trop petit
Essai no 2
Votre proposition : 22
Trop grand
Essai no 3
Votre proposition : 17
Trop grand
Essai no 4
Votre proposition : 16
Bravo ! Vous avez trouvé 16 en 4 essais
```

Programmer en python ce jeu !

A3

Reprendre l'approximation de Pi selon **Leibnitz** pour répondre aux questions :

1. Calculer le premier terme.
2. Afficher la série par série de 20 termes.
3. Montrer que la série est convergente.
4. Trouver la valeur de Pi avec le plus grand nombre de chiffre après la virgule possible.

$$\pi = 4 \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = 4 \left(\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots \right) = \frac{4}{1 + \frac{1^2}{2 + \frac{3^2}{2 + \frac{5^2}{2 + \dots}}}}$$

A4

On veut tester la simulation d'un dé par un ordinateur. Pour cela on génère dans un tableau des nombres compris entre 1 et 6 représentant les faces.

```
import random
i=input(random.randint(1,6))
print(i)
```

La déclaration des ressources est la suivantes :

algorithme de_test;

type

tab=tableau[1..1000] de entier;

variable

lancer:tablo; (* l'ensemble des lancer *)
nblancer,i,face:**entier**

procedure range_lance

fonction probabilite_face(f:integer):**reel**

En utilisant vos connaissances en probabilités et en python, déterminez la probabilité de chaque face.