Devoir d'Informatique # 1

à rendre le Lundi 03 Septembre 2018, 23h59, au plus tard

Lycée Chaptal

2 BCPST A/B

Version du 21 juin 2018, 11 :43

Le devoir est à rendre sous forme d'un fichier `.py` *à l'adresse mail jona.harter.maths@gmail.com, avant la date indiquée. Il devra présenter l'en-tête ci-dessous que vous complèterez avec les bons renseignements.*
Par ailleurs, vous êtes libres d'organiser votre travail comme vous le souhaitez : notamment vous avez le droit de créer des fonctions en début de fichier afin de les utiliser dans plusieurs exercices ensuite.
Enfin, toute question posée à l'adresse mail ci-dessus pendant les vacances est la bienvenue.

```
#**************************************************
# 2BIOA ou B DM de vacances 2018
# Nom/Prénom : à compléter (écrit en python 3.x)
#**************************************************
Vos réponses ici.
```

EXERCICE 1

Soit $(u_n)_{n\in\mathbf{N}}$ la suite définie par : $\begin{cases} u_0 = 0,\ u_1 = 1,\ u_2 = 2, \\ u_{n+3} = 2\,u_{n+2} + 3\,u_{n+1} - 2\,u_n\,,\ \forall\, n \in \mathbf{N}, \end{cases}$ et $(\mathrm{R}_n)_{n\in\mathbf{N}^*}$ la suite définie par : $\mathrm{R}_n = \dfrac{u_{n+1}}{u_n}$, $\forall\, n \in \mathbf{N}^*$.

Q1) Ecrire une fonction `ListeU(N)` renvoyant la liste des `N` premières valeurs de la suite $(u_n)_{n\in\mathbf{N}}$. Représenter graphiquement les 20 premières valeurs de la suite $(\mathrm{R}_n)_{n\in\mathbf{N}}$.

Q2) Tracer le graphe de la fonction polynomiale $\mathrm{P}(x) = x^3 - 2x^2 - 3x + 2$ sur l'intervalle $[-2, 4]$. Remarquer alors que P admet trois racines réelles λ, μ, ν telles que $|\lambda| > |\mu| > |\nu|$.

Q3) On admet qu'il existe $(a, b, c) \in \mathbf{R}^3$ tel que : $\forall\, n \in \mathbf{N}, \quad u_n = a\,\lambda^n + b\,\mu^n + c\,\nu^n$.

Q3.1. [à rendre sur feuille séparée] Démontrer que la suite $(\mathrm{R}_n)_{n\in\mathbf{N}}$ converge vers λ en cherchant un équivalent de $(u_n)_{n\in\mathbf{N}}$.

Q3.2. En déduire, avec `Python`, une valeur approchée de λ. On considèrera la valeur *bonne* pour un n assez grand tel que $|\mathrm{R}_{n+1} - \mathrm{R}_n| < 10^{-5}$.

Q3.3. [Bonus] Il existe un lien important entre $(u_n)_{n\in\mathbf{N}}$ et le polynôme P : analysez ce lien.

Considérons à présent le polynôme $\mathrm{Q}(x) = x^3\mathrm{P}\left(\dfrac{1}{x}\right) = 1 - 2x - 3x^2 + 2x^3$. Quelles sont les racines de Q ? Quelle est la plus grande à présent ?

En formant une nouvelle suite récurrente $(v_n)_{n\in\mathbf{N}}$, associée au polynôme Q, appliquer la même stratégie pour déterminer une valeur approchée de ν.

Q3.4. [Bonus] En déduire une valeur approchée à à 10^{-5} près de μ à partir des deux questions précédentes.

{NUME}-{2}

EXERCICE 2 – Calculs de probabilités de figures du jeu de Poker

On considère un jeu de 32 cartes. Il est formé de couples de 8 valeurs ordonnées, `valeurs=["7", "8", "9", "10", "V", "D", "R", "A"]`, et de 4 « *couleurs* » , `couleurs=["T", "K", "C", "P"]` (Trèfle , Carreau, Coeur, Pique). On distribue au hasard une « *main* », c'est-à-dire 5 cartes distinctes, et on s'intéresse à des mains particulières :

- ⋆ les « *couleurs* » (5 cartes de même « *couleur* ») ;
- ⋆ les « *quintes* » (5 cartes de valeurs qui se suivent) ;
- ⋆ les « *quintes floches* » (5 cartes de même « *couleur* » et de valeurs qui se suivent).

Q1) Construire une fonction `CARTES` des 32 cartes à partir des deux listes `valeurs` et `couleurs`, chaque carte étant représentée par la paire [*valeur*, *couleur*].

INDICATION ► On pourra commencer par créer une liste vide, que l'on remplit à l'aide d'une boucle `for`.

Q2) Vérifier que le nombre de cartes obtenu est correct.

Q3) Écrire une fonction `TIRERMAIN` sans paramètre qui renvoie une liste de cinq cartes distinctes tirées au hasard.

INDICATION ► On pourra utiliser la fonction `sample` du module `random`. Essayez de voir ce que donne :

```python
import random
random.sample([1,2,3],2)
```

Q4) Écrire une fonction `estCOULEUR` d'argument une main, et qui renvoie un booléen indiquant si cette main est une « *couleur* » ou pas.

Q5) Créer une liste `quintes` de toutes les suites possibles de cinq valeurs d'une quinte.

Q6) En déduire une fonction `estQuinte` d'argument une main et qui renvoie un booléen indiquant si cette main est une « *quinte* » ou pas.

INDICATION ► Pour comparer deux listes de valeurs indépendamment de leur ordre, on pourra utiliser la fonction `sorted`. Testez `sorted([1,3,2])`.

Q7) En déduire ensuite une fonction `estQuinteFloche` d'argument une main et qui renvoie un booléen indiquant si cette main est une « *quinte* » ou pas.

Q8) À partir de 50000 tirages aléatoires de mains, estimer la probabilité d'obtenir une « *couleur* », celle de tirer une « *quinte* » et celle de gagner une « *quinte floche* ».

{ALGO}{2}