Lycée Internationale de Valbonne **2020-2021**
T.D. d'informatique ECE 2

Exercice 1.
Un jeu vidéo comporte N phases de jeu : niveau 1, niveau 2, ... niveau N. On suppose que N est un entier au moins égal 3. Le jeu commence au niveau 1. Ensuite, il faut réussir un niveau pour passer au suivant et ainsi de suite jusqu'au dernier niveau.

Le jeu s'arrête si l'on a échoue à l'un des niveaux ou si l'on a réussi tous les niveaux.

On suppose en outre que, lorsqu'on parvient au niveau k ($k = 1, 2, ..., N$), la probabilité de réussir ce $kème$ niveau est égale $1/k$.

On désigne par X_N la variable aléatoire suivante :

"Nombre de niveaux entièrement franchis au moment où le jeu s'arrête".

Ainsi, pour $k = 1, 2, ..., N - 1$, l'événement $(X_N = k)$ signifie que l'on a échoué au niveau $k + 1$, et l'événement $(X_N = N)$ que l'on est vainqueur du jeu.

1. Compléter la fonction suivante pour quelle simule cette expérience

```
function nf=jeux(N)
        nf=???????
        while (rand()<1/(nf+1))& (?????????) then
                ?????????????
        end
endfunction
```

2. On fixe $N = 10^7$. En utilisant la fonction précédente écrire un programme qui calcule une moyenne du nombre de niveaux franchis.

Exercice 2.
On dispose de trois pièces : une pièce numérotée 0, pour laquelle la probabilité d'obtenir "pile" vaut $\dfrac{1}{2}$ et celle d'obtenir "face" vaut également $\dfrac{1}{2}$, une pièce numérotée 1, donnant "pile" à coup sûr et une troisième pièce, numérotée 2, donnant "face" à coup sûr.
On choisit l'une de ces pièces au hasard et on la lance indéfiniment.
On considère la variable aléatoire X, égale au rang d'apparition du premier "pile" et la variable aléatoire Y , égale au rang d'apparition du premier "face". On convient de donner à X la valeur 0 si l'on n'obtient jamais "pile" et de donner à Y la valeur 0 si l'on n'obtient jamais "face".

On rappelle que , pour tout entier naturel m, l'instruction `grand(1,1,'uin',0,m)` renvoie un entier aléatoire compris entre 0 et m (ceci de façon équiprobable).
On décide de coder "pile" par 1 et "face" par 0 .

1. Compléter le script `Scilab` suivant pour qu'il permette le calcul et l'affichage de la valeur prise par la variable aléatoire X lors de l'expérience réalisée dans cet exercice.

```
piece = grand(1,1,'uin',- - -,- - -)
x=1
if piece==0
then lancer=grand(1,1,'uin',- - -,- - -)
        while lancer==0
                lancer=- - -
                x=- - -
        end
else
        if piece==1 then x=- - -
        end
end
disp(x)
```

2. Justifier que le cas où l'on joue avec la pièce numérotée 2 ne soit pas pris en compte dans le script précédent.

Exercice 3.
Soit n un entier naturel non nul et p un réel de $]0;1[$. On pose $q = 1 - p$
On dispose de deux urnes, l'urne U qui contient n boules numérotées de 1 à n et l'urne V qui contient des boules blanches en proportion p.
On pioche une boule au hasard dans U et on note X la variable aléatoire égale au numéro de la boule tirée.
Si X prend la valeur k, on pioche k boules dans V, une par une, avec remise à chaque fois de la boule tirée, et on appelle Y la variable aléatoire égale au nombre de boules blanches obtenues.

On rappelle les commandes Scilab suivantes qui permettent de simuler des variables usuelles discrètes :
grand $(1, 1,$ 'uin'$, a, b)$ simule une variable aléatoire suivant la loi uniforme sur $[a, b]$
grand $(1, 1,$'bin'$, \mathrm{n}, \mathrm{p})$ simule une variable aléatoire suivant la loi binomiale de paramètres n, p
grand $(1, 1,$ 'geom' $, \mathrm{p})$ simule une variable aléatoire suivant la loi géométrique de paramètre p
grand $(1, 1,$ 'poi'$, a)$ simule une variable aléatoire suivant la loi de Poisson de paramètre a.
Compléter le script Scilab suivant afin qu'il permette de simuler les variables X et Y.

n=input ('entrez la valeur de n :')
p=input ('entrez la valeur de p : ')

$X =$ – – – – – – – – – – – – – –
$Y =$ – – – – – – – – – – – – – –

Exercice 4.
Un joueur A dispose d'une pièce qui a la propriété de faire PILE avec la probabilité $\frac{1}{3}$.
Un joueur B dispose d'une pièce qui a la propriété de faire PILE avec la probabilité $p \in]0; 1[$.
Les résultats des lancers de ces pièces seront toujours supposés indépendants. On effectue le jeu suivant :
Les joueurs A et B lancent leur pièce simultanément jusqu'à ce qu'au moins une des deux pièces donne PILE.
Si A et B ont fait PILE simultanément, le jeu s'arrête sans que personne n'ait gagné d'argent.
Sinon, le premier à obtenir PILE s'arrête et l'autre continue ses lancers jusqu'à obtenir PILE également et paye un euro à son adversaire à chacun des lancers de cette série "en solitaire".
Par exemple, si A a obtenu PILE pour la première fois à son 7-ième lancer et si B a obtenu PILE pour la première fois à son 11-ième lancer, c'est B qui doit payer à A la somme de 4 euros.
On note X la variable aléatoire égale au nombre de lancers effectués par le joueur A et Y la variable aléatoire égale au nombre de lancers effectués par le joueur B et $Z = Y - X$.

1. On commence par écrire en scilab une fonction `lancer(p)` qui simule le lancer simultané des joueurs A et B et renvoie 0 si les deux lancers sont identiques et 1 sinon. On rappelle que la commande `rand()` renvoie un réel choisi au hasard dans $[0; 1[$ et ce selon la loi uniforme.
   Recopier et compléter le programme suivant.

```
function t= lancer(p)
        rA=rand() // lancer joueurA
        rB=rand()
        if rA<1/3  && rB<p then
                t=0
        elseif  ........  && ....... then
        ............
        else
        .......
        end
endfunction
```

2. Recopier et Compléter la fonction suivante pour simuler une partie de ce jeu. On utiliser la fonction `lancer` même si on n'a pas répondu à la question.

```
function g=jeu(p,N)
        g=0 //somme donnée par B à A
        for i=.........
                g=g+.........
        end
endfunction
```

Exercice 5 (EML2018).
Deux individus A et B s'affrontent dans un jeu de Pile ou Face dont les règles sont les suivantes :

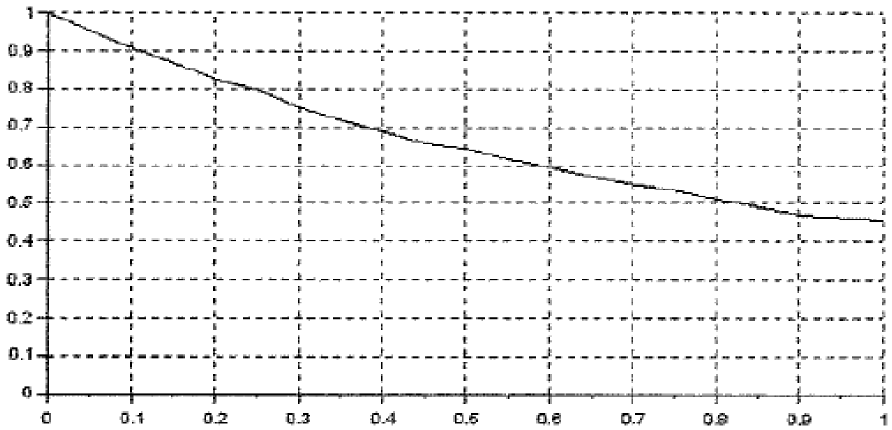
- le joueur A dispose d'une pièce amenant Pile avec la probabilité $\frac{2}{3}$ et lance cette pièce jusqu'à l'obtention du deuxième Pile ; on note X la variable aléatoire prenant la valeur du nombre de Face alors obtenus ;
- le joueur B dispose d'une autre pièce amenant Pile avec la probabilité p et lance cette pièce jusqu'à l'obtention d'un Pile ; on note Y la variable aléatoire prenant la valeur du nombre de Face alors obtenus ;
- Le joueur A gagne si son nombre de Face obtenus est inférieur ou égal à celui de B ; sinon c'est le joueur B qui gagne.

On dit que le jeu est équilibré lorsque les joueurs A et B ont la même probabilité de gagner.

1. Écrire une fonction Scilab d'en-tête `function x = simule_X()` qui simule la variable aléatoire X.
2. écrire une fonction `simule_Y` qui, prenant en argument un réel p de $]0; 1[$, simule la variable aléatoire Y. Expliquer ce que renvoie la fonction suivante :

```
function r = mystere(p)
        r = 0
        N = 10^4
        for k = 1:N
                x = simule_X()
                y = simule_Y(p)
                if x <= y then
                        r = r + 1/N
                end
        end
endfunction
```

3. On trace, en fonction de p, une estimation de la probabilité que A gagne et on obtient le graphe suivant :



À la vue de ce graphe, conjecturer une valeur de p pour lequel le jeu serait équilibré.