ENS LYON Informatique
L3 Probabilités

**Devoir maison numéro 1 : correction**

## Exercice 1 Permutations aléatoires

1. Le plus rapide et élégant (proposé par nombre d'entre vous) est l'algorithme de Knuth ou de Fischer-Yates `https://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle`
2. Par linéarité de l'espérance, on a

$$\mathbf{E}[F(\sigma_n)] = \sum_{k=1}^{n} \mathbf{P}(\sigma_n(k) = k) = 1.$$

## Exercice 2 Bits débiaisés

1. L'algorithme naturel est de produire des paires de bits jusqu'à obtenir une paire de bits distincts, et de retourner `0` ou `1` selon que la paire ainsi produite est `01` ou `10`. Le nombre de paires de bits générées est une variable aléatoire de loi géométrique de paramètre $q := 2p(1-p)$, en particulier il est fini (et l'algorithme termine) presque sûrement.
2. Si on répète l'algorithme précédent, le nombre d'utilisations $N$ de la machine suit la loi de

$$2(X_1 + \cdots + X_n)$$

où les variables aléatoires $(X_i)$ sont i.i.d. de loi géométrique de paramètre $q$. Vous êtes nombreux à avoir majoré $\mathbf{P}(N \geqslant t)$ par l'inégalité de Markov. C'est correct, mais donne un résultat mauvais : on a $\mathbf{P}(N \geqslant tn) \leqslant \frac{1}{tp(1-p)}$ ce qui nécessite $t = \frac{100}{p(1-p)}$. Cela peut être amélioré en utilisant l'inégalité de Tchebychev : en effet $\mathbf{E}N = \frac{n}{p(1-p)}$ et $\mathbf{Var}N = O(n)$ (peu importe la constante exacte, il suffit de dire qu'une v.a. géométrique a une variance finie), donc pour tout $\varepsilon > 0$,

$$\mathbf{P}(N \geqslant (1+\varepsilon)\mathbf{E}N) \leqslant \frac{\mathbf{Var}N}{\varepsilon^2(\mathbf{E}N)^2} = O\left(\frac{1}{\varepsilon^2 n}\right).$$

En particulier, tout réel $t > \frac{1}{p(1-p)}$ convient, une amélioration d'un facteur 100...

Le rendement peut être amélioré car l'algorithme de la question 1 gaspille beaucoup de bits sans en extraire d'information. Par exemple, on peut regarder si les paires de (paires de bits identiques) que l'on a jetées sont identiques ou pas, et déclarer que l'on arrête l'algorithme lorsqu'elles sont différentes, en retournant `0` ou `1` selon la situation `00 11` ou `11 00`. Cela se prête naturellement à une formulation récursive, que vous trouverez expliquée ici `https://en.wikipedia.org/wiki/Bernoulli_process#Randomness_extraction`. On peut montrer que tout $t > 1/h(p)$ convient, où $h(p) = -p\log_2 p - (1-p)\log_2(1-p)$ est la fonction d'entropie binaire, et que cette borne est optimale. Il est remarquable que pour $p$ proche de $1/2$, le rendement obtenu est proche de 1 !

## Exercice 3 Répétitions dans une suite de bits aléatoires

1. Une application directe du principe de linéarité de l'espérance (écrire $N$ comme une somme d'indicatrices...) donne que l'espérance du nombre $N$ de répétitions de longueur $p$ vaut $\mathbf{E}[N] = \frac{n-p+1}{2^{p-1}}$. Lorsque $p = \lfloor \log_2 n \rfloor$, on a $2 + o(1) \leqslant \mathbf{E}[N] \leqslant 4 + o(1)$. De nombreuses copies ont affirmé que $\mathbf{E}[N] \sim 2$, ce qui est incorrect : si $\{\cdot\}$ désigne la partie fractionnaire, la quantitté $2^{\{\log n\}}$ oscille entre 1 et 2...
2. Soit $k = \lfloor n/p \rfloor$. On considère les événements $(A_i)_{0 \leqslant i \leqslant k-1}$ définis par $A_i = \{X_{pi+1} = \cdots = X_{pi+p}\}$. Ces événements sont indépendants (groupement par paquets) et de même probabilité $1/2^{p-1}$. On a

$$\mathbf{P}(\text{au moins une répétition de longueur } p) \geqslant \mathbf{P}\left(\bigcup A_i\right) = 1 - \left(1 - \frac{1}{2^{p-1}}\right)^k \geqslant 1 - \exp(k/2^{p-1}).$$

On vérifie que cette quantité tend vers 1 lorsque $n$ tend vers l'infini dès lors que $p \leqslant 0.99 \log_2 n$.

**Exercice 4 Comment maximimser la probabilité de rencontrer l'âme sœur**

Soit $E$ l'événement où la princesse épouse le prétendant numéro 1, et $A_i$ l'événement où le prétendant numéro 1 lui est présenté en position $i$. Si $i > m$, conditionnellement à $A_i$, l'événement $E$ est réalisé si et si seulement si le mieux classé des $i-1$ premiers prétendants a été présenté parmi les $m$ premiers. Par symétrie, on a donc $\mathbf{P}(E|A_i) = \frac{m}{i-1}$. Il vient

$$\mathbf{P}(E) = \sum_{i=m+1}^{n} \mathbf{P}(E \cap A_i) = \sum_{i=m+1}^{n} \mathbf{P}(A_i)\mathbf{P}(E|A_i) = \frac{m}{n} \sum_{i=m+1}^{n} \frac{1}{i-1}.$$

On utilise l'encadrement $\log(n+1) \leqslant \sum_{k=1}^{n} \frac{1}{k} \leqslant 1 + \log(n)$ pour obtenir

$$\frac{m}{n}(\log(n) - \log(m)) \leqslant \mathbf{P}(E) \leqslant \frac{m}{n}(\log(n-1) - \log(m-1)).$$

En négligeant les approximations par la partie entière, il s'ensuit que $\mathbf{P}(E)$ est maximal pour $m \approx n/e$, et vaut $\mathbf{P}(E) \approx 1/e$.