# Évaluation de travaux pratiques

Durée de l'épreuve : **1h20**.
Le barème est donné à titre indicatif et est susceptible d'être modifié.
Toute solution partielle pourra être prise en compte dans l'évaluation.

**Exercice 1** Lois discrètes **6 points**

1. Écrire une fonction `bernoulli(p,ft)` qui renvoie une matrice de format `ft` dont les coefficients sont des nombres aléatoires indépendants tirés suivant la loi de Bernoulli de paramètre `p`.
2. Écrire une fonction `binomiale(n,p,N)` qui renvoie un vecteur ligne de taille `N` dont les coefficients sont des nombres aléatoires indépendants tirés suivant la loi binomiale de paramètres `n` et `p`.

**Exercice 2** **8 points**

Soit $X$ une variable aléatoire de densité $f_X$ qui est définie par :

$$f_X(x) = \begin{cases} \frac{x}{2} & \text{si } x \in [0\,;2] \\ 0 & \text{sinon} \end{cases}$$

1. Calculer la fonction de répartition $F_X$ de $X$.
2. Justifier que la fonction $G_X$ définie sur $]0\,;1[$ par $G_X(y) = 2\sqrt{y}$ est l'« inverse de la fonction de répartition $F_X$ ».
3. Par la méthode d'inversion de la fonction de répartition, écrire une fonction `inv_rep(N)` qui renvoie un vecteur ligne de taille `N` dont les coefficients sont des nombres aléatoires indépendants de même loi que $X$.
4. Décrire brièvement sur la copie comment l'on peut simuler $X$ par la méthode du rejet.
5. Par la méthode du rejet, écrire une fonction `rejet(N)` qui renvoie un vecteur ligne de taille `N` dont les coefficients sont des nombres aléatoires indépendants de même loi que $X$.

**Exercice 3** **8 points**

On souhaite donner une valeur numérique approchée des intégrales suivantes :

$$I = \int_0^1 (1+x)^4 \sin(x^2+1)\,\mathrm{d}x$$
$$J = \int_0^\infty (1+x)^4 \sin(x^2+1) e^{-2x}\,\mathrm{d}x$$

1. Décrire brièvement sur la copie comment l'on peut trouver une valeur approchée de $I$ par la méthode de Monte-Carlo.
2. Écrire un programme `montecarloI.m` qui renvoie une valeur approchée de $I$ en faisant un tirage de 10000 nombres aléatoires.
3. Décrire brièvement sur la copie comment l'on peut trouver une valeur approchée de $J$ par la méthode de Monte-Carlo.
4. Écrire un programme `montecarloJ.m` qui renvoie une valeur approchée de $J$ en faisant un tirage de 10000 nombres aléatoires.