1. Sea $g\colon [0,1] \to \mathbb{R}$ y $\theta = \int_0^1 g(x)\,dx$. Queremos aproximar θ mediante un método de Montecarlo. Sean $U_1, \ldots, U_n, \ldots$ varables aleatorias i.i.d con distribución uniforme en $[0,1]$ y $\theta_n := \text{lím}_{n\to\infty} \frac{1}{n}\sum_{i=1}^n g(U_i)$. Como
$$\theta = E(g(U_1)) = \lim_{n\to\infty} \hat{\theta}_n,$$
utilizamos este hecho para aproximar.

   a) Probar que $\hat{\theta}_n$ es insesgado y fuertemente consistente (i.e. $E(\hat{\theta}_n) = \theta$ y $\hat{\theta}_n \to \theta$ casi seguramente).
   b) Implementar un algoritmo que estime θ cuando $g(x) = x^2$.
   c) Implementar un algoritmo que grafique el error $\hat{\theta}_n - \theta$.
   d) Idem para $g(x) = \cos(1/x)$.
   e) Idem para $g(x_1, x_2) = 4x_1^2 x_2 + x_2^2$. Sugerencia:

```
>x=rand(2,n);
>g=4*(x(1,:).^2).*x(2,:) + x(2,:).^2;
>ans=mean(g)
```

2. Utilizando el comando `rand`, implementar un agloritmo que simule una variable aleatoria con distribución

   a) $\mathcal{U} \sim [\pi, 2\pi]$
   b) $\mathbb{E}(\lambda)$.
   c) $\Gamma(8, \lambda)$
   d) $N(0,1)$ y $N(\mu, \sigma^2)$ (Sug.: recordar el método de Box-Muller o el TCL).
   e) $\beta(3,5)$
   f) Bernoulli
   g) $B(20, 0{,}02)$
   h) Poisson
   i) Geométrica
   j) Un vector (X, Y) con marginales Normales estándar y $\mathrm{Cov}(X,Y) = \pi$.

   Sugerencia: No usar **siempre** el método de la inversa generalizada. Wikipedia puede servir.

   En los casos en que se le ocurra más de un método, usar los comandos tic y toc para comparar la velocidad de cada uno.

3. Estadísticos de oren. Sean $X_1, \ldots, X_n$ variables aleatorias i.i.d con distribución exponencial de parámetro λ. Sean $X_{(1)}, \ldots, X_{(n)}$ los estadísticos de orden. Implementar un algoritmo que simule $X_{(i)}$.

   a) Utilizar el primer método que se le ocurra.
   b) ¿Se puede hacer mejor? (Recordar que las funciones de distribución son monótonas)
   c) ¿Se puede hacer mejor? (Recordar la distribución Beta)

4. Método de aceptación-rechazo.

   a) Sean $A \subset B \subset \mathbb{R}^d$. Probar que el siguiente algoritmo genera una variable aleatoria con distribución uniforme en B.

generar Z uniforme en B
mientras $Z \notin A$
    generar Z uniforme en B
fin mientras
salida=Z

b) Sea τ la cantidad de iteraciones que hace el algoritmo antes de parar. Hallar $E(\tau)$. ¿En qué situaciones es conveniente usar este algoritmo?

c) Consideremos una densidad f soportada en $[0,1]$ y el conjunto $G_f := \{(x,y)\colon 0 \leq y \leq f(x)\}$. Sea (X,Y) un vector con distribución uniforme en G_f. Probar que X tiene densidad f.

d) Implementar un algoritmo para simular una variable $X \sim \beta(\pi, e)$
Sugerencia: Acá va la función `dens_triang` que devuelve una muestra con densidad $f(x) = 2x\mathbf{1}_{[0,1]}(x)$ y un script que hace una histograma de una muestra de tamaño n.

```
function resp=dens_triang;
x=rand(2,1);
while x(2)>x(1)
x=rand(2,1);
end
resp=x(1);
```

```
n=1000;
x=zeros(1,n);
for j=1:n
x(j)=dens_triang;
end
hist(x)
```

5. Composición. Implementar un algoritmo para simular una variable aleatoria con densidad

$$f(x) = \frac{\lambda_1}{3}e^{-\lambda_1 x} + \frac{4\lambda_2}{9}e^{-\lambda_2 x} + \frac{2\lambda_3}{9}e^{-\lambda_3 x} \quad (x > 0).$$

6. Implementar un algoritmo para estimar

$$\int_0^\infty \int_0^\infty \int_0^\infty \sin(xyz) 2e^{-(x+2y+z)}\, dxdydz$$

Sugerencia: Si (X,Y) tiene densidad f, ¿cómo se calcula $E(g(X,Y))$? Opcional: comparar con otros métodos de cuadraturas. En MATLAB/Octave usar `triplequad`.