Consider

$$\alpha = \int_0^1 h(x)dx$$

where

$$h(x) = 100 \cdot I_{(0,1/100]}(x) + 1 \cdot I_{(1/100,1)}(x).$$

- Implement Pseudocode for omc_integral(n):
  - Generate n iid samples
  $$\{iid\ Y_i \sim U(0,1) : i = 1, 2, \ldots, n\};$$
  - Compute n X samples by
  $$\{X_i = h(Y_i) : i = 1, 2, \ldots, n\};$$
  - Take average of X_i's
- Demonstrate convergence rate numerically by doing the following:
  - Fix a batch number $m = 100$;
  - For i in range(5, 10):
    * run m times of omc_integral($n = 2^i$), store it into $\{\alpha_{ij} : j = 1, \ldots m\}$.
    * compute standard deviation (*numpy.std*) of $\{\alpha_{ij} : j = 1, \ldots m\}$, save it to σ_i.
  - plot and find slope (scipy.stats.linregress) for the data
  $$\{(i, -\log_2 \sigma_i) : i = 5, \ldots, 10\}.$$
- Find its convergence rate by the following procedure:
  Compute RMSE (root MSE) for $\hat{\alpha}_n$ in terms of $Cn^{-\alpha}$. We say α as the convergence rate.