Warm-up

Problem 1. If you repeatedly perform an experiment whose outcome has a Geometric Distribution with probability of success p, what is the expected number of times you need to repeat the experiment before you get a success? $\frac{1}{p}$

Problem solving

Problem 2. Suppose you only have access to a biased coin that lands heads with probability p and tails with probability $(1-p)$. Show how to design a fair coin without even knowing what p is. How many times does your algorithm flip the biased coin in expectation?

Problem 3. Suppose you only have access to a fair coin that lands heads with probability $1/2$ and tails with probability $1/2$. Show how to design an algorithm for uniformly sampling an integer from $\{1, \ldots, n\}$. How many times does your algorithm flip the coin in expectation?

Problem 4. Suppose you only have access to a random generator for sampling real numbers from the interval $[0, 1]$. Show how to design an algorithm for uniformly sampling points (x, y) from the square $[-1, 1]^2$ (i.e., $-1 \leq x \leq 1$ and $-1 \leq y \leq 1$). How many samples does your algorithm need in expectation?

Problem 5. Suppose you only have access to a random generator for sampling real numbers from the interval $[0, 1]$. Show how to design an algorithm for uniformly sampling points (x, y) from the unit radius disk centered at the origin (i.e., $x^2 + y^2 \leq 1$). How many samples does your algorithm need in expectation?

Problem 6. Suppose we want to add a new operation to the existing skip list implementation. This operation, RangeSearch($k1$, $k2$), returns all the items with keys in the range $[k1, k2]$. Design this operation and show that it runs in expected time $O(\log n + s)$, where n is the number of elements in the skip list and s is the number of items returned.

1. $\frac{1}{p}$

2. $\left.\begin{array}{l} p^2 \to HH \\ (1-p)^2 \to TT \end{array}\right\}$ ignore these

$\left.\begin{array}{l} p(1-p) \to HT \\ (1-p)p \to HT \end{array}\right\}$ same

Expected throw:

$$2 \times \frac{1}{p(1-p)+(1-p)p}$$

since each pair we throw 2 coins

Problem 3. Suppose you only have access to a fair coin that lands heads with probability 1/2 and tails with probability 1/2. Show how to design an algorithm for uniformly sampling an integer from $\{1, \ldots, n\}$. How many times does your algorithm flip the coin in expectation?

① using binary number

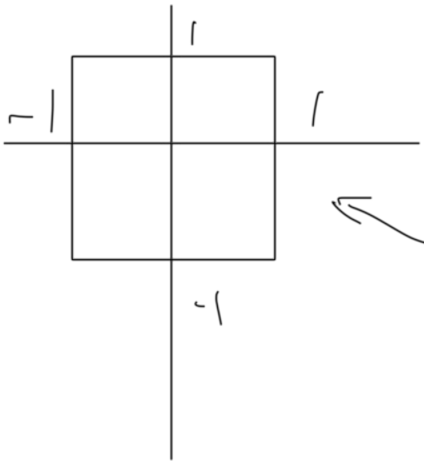
_ _ _ _ _ _

each binary number correspond to a number

$\lfloor \log n \rfloor + 1$ for each time

Problem 4. Suppose you only have access to a random generator for sampling real numbers from the interval $[0,1]$. Show how to design an algorithm for uniformly sampling points (x,y) from the square $[-1,1]^2$ (i.e., $-1 \le x \le 1$ and $-1 \le y \le 1$). How many samples does your algorithm need in expectation?

$r = \text{random } [0, 1]$



$x = 2r - 1$

$y = 2r - 1$

Problem 5. Suppose you only have access to a random generator for sampling real numbers from the interval $[0, 1]$. Show how to design an algorithm for uniformly sampling points (x, y) from the unit radius disk centered at the origin (i.e., $x^2 + y^2 \leq 1$). How many samples does your algorithm need in expectation?