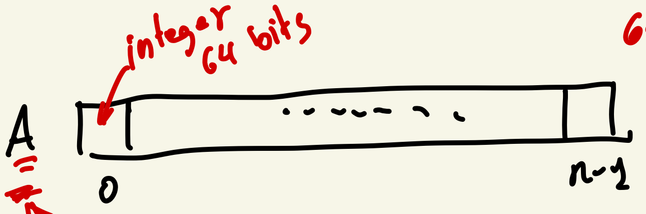
Lecture 1 : Introduction

- The course is "algorithms".

- Golden standard for classical algorithms:

n input size To acheive O(n) time and space. O(n)

64 bits × n = 64n bits

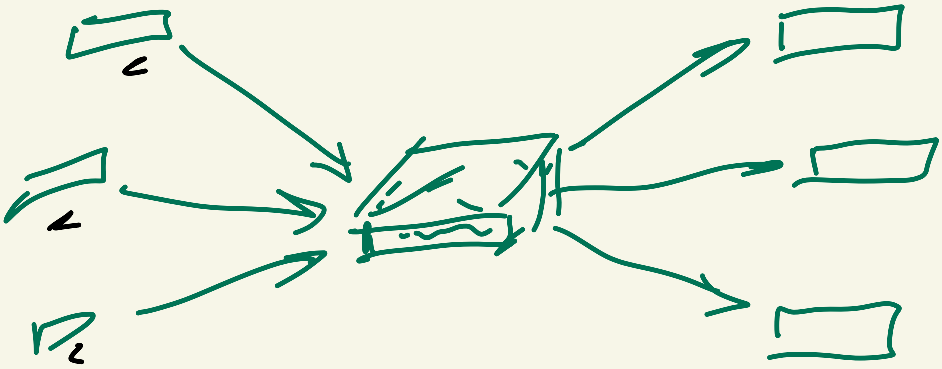


- Massive data : The data we work with has grown much larger than the computer resources we have.

- Algorithms for massive data :
    - To think of algorithms which limits the access to data.
    - These limitations can be either limited time and/or limited space

- Common scenario : limited space :
    - Suppose we have a router with interet traffic passing through it.
    - The router recieves information along with a soure and IP address, and sends it to the corresponding address.

- Let's say we would like to compute certain statistic with the information passing through the router.



- For example, we would like to know the # of times a certain IP address makes a request.

* The amount of information passing through the router greatly exceeds its available storage.
"limited space"

. Therefore, we cannot simply store copies of data, and then compute based on the stored data.

- Many of these tasks are impossible to solve by classical computation/algorithms.

New way of thinking: In order to solve the problems, we relax the guarantees

- New way of thinking: In order to solve the problems, we relax the guarantees
  - Instead of producing exact answers, we need ways to approximate it.
  - Instead of requiring the approximations to work all the time, we require that they work with high probability of success.
- One way to approximate is to say:

  true answer $\leq$ output $\leq \alpha *$ true answer

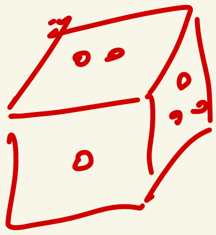
  Here α is the approximation ratio, and $\alpha = 1 + \epsilon$, where ϵ is very small.

- One way to give confidence of output is to say that

  "the approximation ratio α holds with probability $1 - \delta$."

  where δ as being a very small. (the probability is close to 1.)

P.M.F.



Fair dice

$$X = \begin{cases} 1 & \text{with prob. } 1/6 \\ 2 & \text{〃} \\ 3 & \text{〃} \\ 4 & \text{〃} \\ 5 & \text{〃} \\ 6 & \text{〃} \end{cases}$$

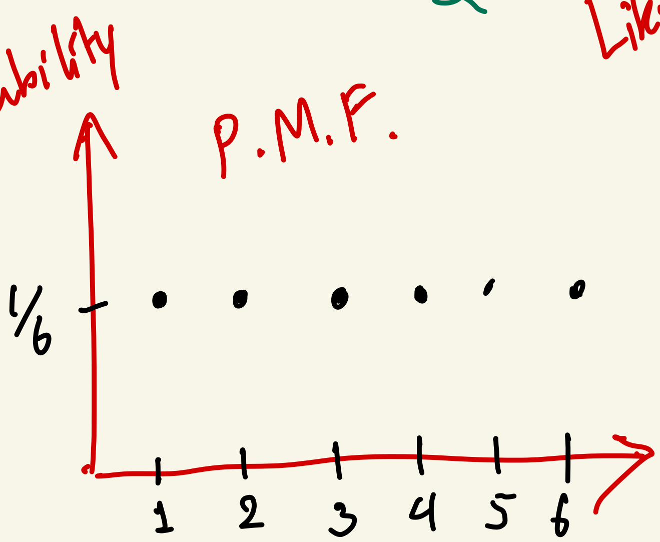
$$\boxed{E[X] = \frac{1+2+3+4+5+6}{6} = \frac{21}{6} \approx 3.4}$$

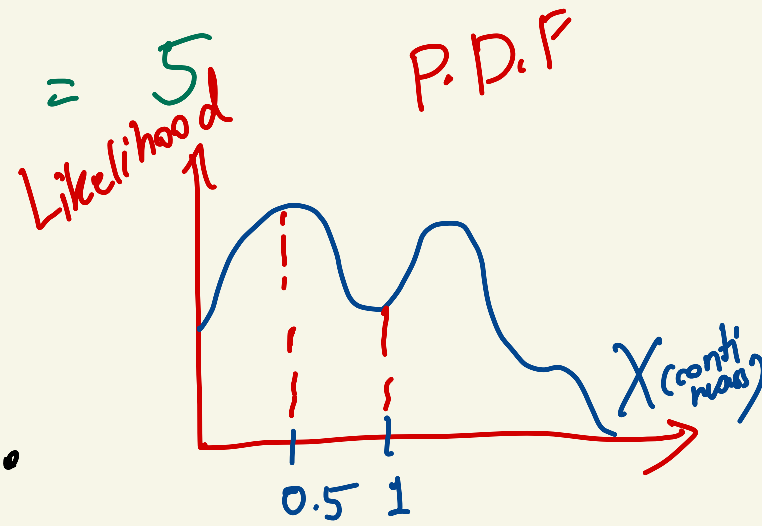
H T

Fair coin

$$Y = \begin{cases} 0 & \text{with prob. } 1/2 \text{ (ออกก้อย)} \\ 10 & \text{with prob. } 1/2 \text{ (ออกหัว)} \end{cases}$$

$$E[Y] = \frac{0+10}{2} = 5$$





Probability review: There are a few probability tools we will use in the analysis of algorithms in this class.

In the following, we let x be a random variable.

- Def 1 (Expectation):
  - For a discrete r.v. X, the expectation of x, $\mathbb{E}[X]$ is

$$\mathbb{E}[X] = \sum_a a \Pr\{x = a\}$$

Ex. $\mathbb{E}[X] = 1 \times \Pr\{x = 1\} + 2 \cdot \Pr\{x = 2\} + \dots + 6 \cdot \Pr\{x = 6\}$

$$= \left(1 \times \frac{1}{6}\right) + \left(2 \times \frac{1}{6}\right) + \dots + \left(6 \times \frac{1}{6}\right)$$

$$= \frac{1}{6}(1 + 2 + \dots + 6) = \frac{21}{6} = 3.4x$$

- For a continuos r.v. X, the expectation of x, $\mathbb{E}[X]$ is

$$\mathbb{E}[X] = \int a\, \phi(a)\, da$$

where ϕ is the probability density function of x.

- <u>Lemma 2 (Linearity of Expectation)</u>.

Let X and Y be two r.v.s. Then,

$$\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$$

- <u>Lemma 3 (Markov's inequality)</u>.

Let X be a non-negative r.v. Then, $\forall \lambda > 0$

$$\Pr\{X > \lambda\} \leq \frac{\mathbb{E}[X]}{\lambda}$$

<u>Ex.</u> for any r.v. $X > 0$, we get

$$\Pr\{X > \underbrace{10\,\mathbb{E}[X]}_{\lambda}\} \leq \frac{\mathbb{E}[X]}{10\,\mathbb{E}[X]} = \frac{1}{10}$$

- With 10% confidence, we guarantee that X will deviate from its expectation by more than a factor of 10.
- Equivalently, with 90% confidence, we guarantee that X will deviate from its expectation within a factor of 10.

<u>Def 4 (Variance)</u>. The variance of a r.v. X, denoted $\mathrm{Var}[X]$, is

$$\mathrm{Var}[X] = \mathbb{E}\left[(X-\mathbb{E}[X])^2\right] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$$

<u>Lemma 5 (Chebyshev's Inequality)</u>.

$$\forall \lambda > 0, \quad \Pr\{|X-\mathbb{E}[X]| > \lambda\} \leq \frac{\mathrm{Var}[X]}{\lambda^2}$$

<u>E.x.</u> by setting $\lambda^2 = 10\,\mathrm{var}[X] > 0$,

we can find that

$$\Pr\{|X-\mathbb{E}[X]| > \underbrace{\pm\sqrt{10\,\mathrm{var}[X]}}_{\lambda}\} \leq \frac{\mathrm{var}[X]}{10\,\mathrm{var}[X]} = \frac{1}{10}$$

$$\Longleftrightarrow \Pr\{\underbrace{|X-\mathbb{E}[X]| \leq \sqrt{10\,\mathrm{var}[X]}}_{X = \mathbb{E}[X] \pm \sqrt{10\,\mathrm{var}[X]}}\} \leq \frac{9}{10}$$

An Easy Problem : Counting

- Say that we are maintaining a router that counts the # of packets that has been received.

- Surely to keep track of up to n different packets, our router should have a memory of at least $\log_2 n$ bits.
- Suppose n is very large that $\log_2 n$ is still a large number.
(This means we cannot implement the counter)
- We will do better with approximation and randomization (Morris algorithm)

Morris algorithm (1978)

- An algorithm to approximately compute the value of n by using $O(\log\log n)$ bits only.

Algorithm

1. Set a counter variable X, initialize X to 0 ($X \leftarrow 0$).

2. For each new packet, we increment X by 1 with probability $\frac{1}{2^X}$; otherwise, X remains unchanged.

3. For a query for the # of packets received, return the estimator

$$E = 2^X - 1$$

For analysis of the algorithm, we denote X_k the value of X after k increments; $E_k = 2^{X_k} - 1$. Note $X_2, \ldots, X_n$ and $E_1, \ldots, E_n$ are R.V.s.

Extra: induction

Claim 1. $\mathbb{E}[E_n] = n$.

Proof. - We use induction on n.

- Base case ($n=0$):

$\Rightarrow X_0 = 0$

$E_0 = 2^{X_0} - 1 = 2^0 - 1 = 1 - 1 = 0$

$\Rightarrow \mathbb{E}[E_0] = 0 = n$

Therefore, the base case is true.

- Inductive step:

- Assume by induction that

$\mathbb{E}[E_{n-1}] = n-1$

we will show that $\mathbb{E}[E_n] = n$.

- Note that $\mathbb{E}[E_n] = n$ means

$$\mathbb{E}[E_n] = \sum_{i=0}^{i} (2^i - 1) \times \Pr\{X_n = i\} = n.$$

$$= \sum_{i=1} (2^i - 1) \times \Pr\{X_n = i\} = n.$$

$$\Pr\{X_n = i\} = \Pr\{X \text{ is incremented and } X_{n-1} = i-1\} + \Pr\{X \text{ is not incremented and } X_{n-1} = i\}$$

$$= \frac{1}{2^{i-1}} \times \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \Pr\{X_{n-1} = i\}$$

$$\mathbb{E}[E_n] = \sum_{i=1} (2^i - 1) \times \Pr\{X_n = i\}$$

$$= \sum_{i=1} (2^i - 1)\left(\frac{1}{2^{i-1}} \cdot \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \cdot \Pr\{X_{n-1} = i\} \right)$$

$$\mathbb{E}[2^{X_n}]$$

$$\| $$

$$\sum_{i=1} 2^i \left(\frac{1}{2^{i-1}} \cdot \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \cdot \Pr\{X_{n-1} = i\} \right)$$

$$= \sum_{i=1} \left(2 \cdot \Pr\{X_{n-1} = i-1\} + 2^i \cdot \Pr\{X_{n-1} = i\} - \Pr\{X_{n-1} = i\} \right)$$

$$= 2 \sum_{i=1} \Pr\{X_{n-1} = i-1\} + \sum_{i=1} 2^i \cdot \Pr\{X_{n-1} = i\} -$$

$$\sum_{i=1} \Pr\{X_{n-i} = i\}$$

$$= 2 \sum_{i=1} \Pr\{X_{n-1} = i-1\} +$$

$$\underbrace{\sum_{i=1} \left(2^i \cdot \Pr\{X_{n-1} = i\} - \Pr\{X_{n-1} = i\} \right)}$$

$$\boxed{\underbrace{\sum_{i=1} (2^i - 1) \cdot \Pr\{X_{n-1} = i\}}_{\mathbb{E}[E_{n-1}] = n-1}}$$

$$\Rightarrow \quad = 2\underbrace{\sum_{i=1} \Pr\{X_{n-1} = i-1\}}_{1} + n-1$$

$$= 2 + n - 1$$

$$= n + 1$$

$$\mathbb{E}[E_n] = \sum_{i=1} (2^i - 1)\left(\frac{1}{2^{i-1}} \cdot \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \cdot \Pr\{X_{n-1} = i\}\right)$$

$$= (n+1) + \underbrace{\sum_{i=1} (-1)\left(\frac{1}{2^{i-1}} \cdot \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \cdot \Pr\{X_{n-1} = i\}\right)}_{-1 \text{ (your homework)}}$$

$$= n \qquad \square$$

Alternative proof:

- Goal: show $\mathbb{E}[E_n] = n$

$$\Rightarrow \mathbb{E}[E_n] = \mathbb{E}[\underbrace{2^{X_n}}_{r.v}+\underbrace{(-1)}_{r.v.}]$$

$$= \underbrace{\mathbb{E}[2^{X_n}]}_{\sum_{i=0} 2^i \Pr\{X_n = i\}} + \underbrace{\mathbb{E}[-1]}_{-1} = n+1-1$$

$$= n$$

$$= \underbrace{\sum_{i=1} 2^i \Pr\{X_n = i\}}_{n+1}$$

$\square$

Claim 2: $\operatorname{Var}[E_n] \leq \frac{3n(n+1)}{2} + 1 = O(n^2)$.

Proof: - Note that

$$\operatorname{Var}[E_n] = \operatorname{var}[2^{X_n} - 1] = \operatorname{var}[2^{X_n}].$$

$$- \operatorname{Var}[2^{X_n}] = \mathbb{E}[(2^{2X_n})] - \underbrace{\left(\mathbb{E}[2^{X_n}]\right)^2}_{\geq 0}$$

$$\Rightarrow \operatorname{Var}[2^{X_n}] \leq \mathbb{E}[(2^{2X_n})].$$

$$- \mathbb{E}[2^{2X_n}] = \sum_{i=0} 2^{2i} \cdot \Pr\{X_n = i\}$$

$$= \sum_{i=0} 2^{2i} \cdot \left(\frac{1}{2^{i-1}} \cdot \Pr\{X_{n-1} = i-1\} + \left(1 - \frac{1}{2^i}\right) \cdot \Pr\{X_{n-1} = i\} \right)$$

$$= \sum_{i=0} \left(2^{i+1} \cdot \Pr\{X_{n-1} = i-1\} + (2^{2i} - 2^i) \cdot \Pr\{X_{n-1} = i\} \right)$$

$$= \sum_{i=0} 2^{i+1} \cdot \Pr\{X_{n-1} = i-1\} + \underbrace{\sum_{i=0} 2^{2i} \cdot \Pr\{n_{n-1} = i\}}_{\mathbb{E}[2^{2X_{n-1}}]} - \underbrace{\sum_{i=0} 2^i \Pr\{X_{n-1} = i\}}_{\mathbb{E}[2^{X_{n-1}}]}$$

$$= \mathbb{E}[2^{2X_{n-1}}] + 4 \sum_{i=0} \underbrace{2^{i-1} \cdot \Pr\{X_{n-1} = i-1\}}_{\mathbb{E}[2^{X_{n-1}}]} - \sum_{i=0} \underbrace{2^i \Pr\{X_{n-1} = i\}}_{\mathbb{E}[2^{X_{n-1}}]}$$

$$= \mathbb{E}[2^{2X_{n-1}}] + 4\,\mathbb{E}[2^{X_{n-1}}] - \underbrace{\sum_{i=0} 2^i \Pr\{X_{n-1} = i\}}_{\mathbb{E}[2^{X_{n-1}}]}$$

$$= \mathbb{E}[2^{2X_{n-1}}] + 3\,\mathbb{E}[2^{X_{n-1}}]$$

$$\Rightarrow \mathrm{Var}[E_n] \leq \mathbb{E}[2^{2X_n}] = \mathbb{E}[2^{2X_{n-1}}] + 3\,\mathbb{E}[2^{X_{n-1}}]$$

$\therefore$ So by induction and noting that

$$\mathbb{E}[2^{2X_0}] = \mathbb{E}[2^{2(0)}] = \mathbb{E}[2^0] = 1,$$

it follows that

$$\mathbb{E}[2^{2X_n}] = 3\,\mathbb{E}[2^{X_{n-1}}] + \underbrace{\mathbb{E}[2^{2X_{n-1}}]}_{3\mathbb{E}[2^{X_{n-2}}] + \underbrace{\mathbb{E}[2^{2X_{n-2}}]}_{3\mathbb{E}[2^{X_{n-3}}] + \mathbb{E}[2^{2X_{n-3}}]}}$$

$$\Rightarrow \mathbb{E}[2^{2X_n}] = 3 \sum_{i=0}^{n-1} \mathbb{E}[2^{X_i}] + 1$$

$$\Rightarrow \mathbb{E}\left[2^{2X_n}\right] = 3\sum_{i=0}^{n-1} \underbrace{\mathbb{E}\left[2^{X_i}\right]}_{i+1} + 1$$

$$= 3\sum_{i=0}^{n-1} i+1 \quad + 1$$

$$= 3\,(\underbrace{1+2+3+\ldots+n}_{\frac{n(n+1)}{2}}) + 1$$

$$= \frac{3\,n(n+1)}{2} + 1$$

$$\operatorname{var}[E_n] \leq \mathbb{E}\left[2^{2X_n}\right] = \frac{3\,n(n+1)}{2} + 1$$

$$\square$$

- From Claim 1, $\mathbb{E}[E_n] = n$.
- From Claim 2, $\text{Var}[E_n] \leq \frac{3n(n+1)}{2} + 1$.

<u>Recall</u>: Chebyshev's Inequality,

$$\forall \lambda > 0, \ \Pr\{|X - \mathbb{E}[X]| > \lambda\} \leq \frac{\text{Var}[X]}{\lambda^2}$$

$$\Rightarrow \forall \lambda > 0, \ \Pr\{|E_n - \mathbb{E}[E_n]| > \lambda\} \leq \frac{\text{Var}[E_n]}{\lambda^2}$$

Suppose we want $\leq 10\%$ failure probability.

$$\Rightarrow \Pr\{|E_n - \mathbb{E}[E_n]| > \lambda\} \leq 10\%$$

$$\Rightarrow \frac{\text{Var}[E_n]}{\lambda^2} = \frac{1}{10}$$

$$\Rightarrow \lambda^2 = 10 \,\text{Var}[E_n] = 10 \times \underbrace{\left(\frac{3n(n+1)}{2} + 1\right)}_{\Theta(n^2)}$$

$$\Rightarrow \lambda = \sqrt{10\, \Theta(n^2)}$$

$$\approx (3.3)\, \Theta(n) = \Theta(n)$$

$\approx 3.3n$

$$\Rightarrow E_n = \mathbb{E}[E_n] \pm \lambda = n \pm \Theta(n)$$

with probability 90%.

- To get a better result, we use a general technique in order to maintain the mean (expectation) and reduce the variance, which is to run multiple instances of Morris algorithm independenty and take the average.

Morris+ Algorithm:

1. Run k copies of the basic Morris algorithm. Denote the x values in the i-th run $X^{(i)}$.

2. Upon a query, return the estimator

$$E = \frac{1}{k} \sum_{i=1}^{k} \left(2^{x(i)} - 1 \right).$$

Claim 3: $\mathbb{E}[E] = n$.

Claim 4: $\mathrm{Var}[E] \leq \frac{3}{2} \frac{n^2}{k}$.