# CSE 548: Analysis of Algorithms

# Lectures 21 – 23 & 26
# ( Randomized Algorithms
# & High Probability Bounds )

**Rezaul A. Chowdhury**

**Department of Computer Science**
**SUNY Stony Brook**
**Spring 2015**

# Markov's Inequality

**Theorem 1:** Let $X$ be a random variable that assumes only nonnegative values. Then for all $\delta > 0$,

$$\Pr[X \geq \delta] \leq \frac{E[X]}{\delta}.$$

**Proof:** For $\delta > 0$, let

$$I = \begin{cases} 1 & \text{if } X \geq \delta; \\ 0 & \text{otherwise.} \end{cases}$$

Since $X \geq 0$, $I \leq \frac{X}{\delta}$.

We also have, $E[I] = \Pr[I = 1] = \Pr[X \geq \delta]$.

Then $\Pr[X \geq \delta] = E[I] \leq E\left[\frac{X}{\delta}\right] \leq \frac{E[X]}{\delta}$.

# Example: Coin Flipping

Let us bound the probability of obtaining more than $\frac{3n}{4}$ heads in a sequence of $n$ fair coin flips.

Let

$$X_i = \begin{cases} 1 & \text{if the } i\text{th coin flip is heads;} \\ 0 & \text{otherwise.} \end{cases}$$

Then the number of heads in $n$ flips, $X = \sum_{i=1}^{n} X_i$.

We know, $E[X_i] = \Pr[X_i = 1] = \frac{1}{2}$.

Hence, $E[X] = \sum_{i=1}^{n} E[X_i] = \frac{n}{2}$.

Then applying Markov's inequality,

$$\Pr\left[X \geq \frac{3n}{4}\right] \leq \frac{E[X]}{3n/4} = \frac{n/2}{3n/4} = \frac{2}{3}.$$

# Chebyshev's Inequality

**Theorem 2:** For any $\delta > 0$,

$$\Pr[|X - E[X]| \geq \delta] \leq \frac{Var[X]}{\delta^2}.$$

**Proof:** Observe that $\Pr[|X - E[X]| \geq \delta] = \Pr[(X - E[X])^2 \geq \delta^2]$.

Since $(X - E[X])^2$ is a nonnegative random variable, we can use Markov's inequality,

$$\Pr[(X - E[X])^2 \geq \delta^2] \leq \frac{E\left[(X-E[X])^2\right]}{\delta^2} = \frac{Var[X]}{\delta^2}.$$

# Example: $n$ Fair Coin Flips

$$X_i = \begin{cases} 1 & \text{if the } i\text{th coin flip is heads;} \\ 0 & \text{otherwise.} \end{cases}$$

Then the number of heads in $n$ flips, $X = \sum_{i=1}^{n} X_i$.

We know, $E[X_i] = \Pr[X_i = 1] = \frac{1}{2}$ and $E[(X_i)^2] = E[X_i] = \frac{1}{2}$.

Then $Var[X_i] = E[(X_i)^2] - (E[X_i])^2 = \frac{1}{2} - \frac{1}{4} = \frac{1}{4}$.

Hence, $E[X] = \sum_{i=1}^{n} E[X_i] = \frac{n}{2}$ and $Var[X] = \sum_{i=1}^{n} Var[X_i] = \frac{n}{4}$.

Then applying Chebyshev's inequality,

$$\Pr\left[X \geq \frac{3n}{4}\right] \leq \Pr\left[|X - E[X]| \geq \frac{n}{4}\right] \leq \frac{Var[X]}{(n/4)^2} = \frac{n/4}{(n/4)^2} = \frac{4}{n}.$$

# Coin Flipping and Randomized Algorithms

Suppose we have an algorithm that is

- correct ( heads ) only with probability $p \in (0,1)$, and
- incorrect ( tails ) with probability $1 - p$.

**Question:** How many times should we run the algorithm to be reasonably confident that it returns at least one correct solution?

- Las Vegas Algorithm: You keep running the algorithm until you get a correct solution. What is the bound on running time?

- Monte Carlo Algorithm: You stop after a certain number of iterations no matter whether you found a correct solution or not. What is the probability that your solution is correct ( or you found a solution )?

# Coin Flipping and Randomized Algorithms

Suppose we have an algorithm that is

- correct ( heads ) only with probability $p \in (0,1)$, and
- incorrect ( tails ) with probability $1 - p$.

Suppose we run the algorithm $k$ times.

Then probability that no run produces a correct solution is $(1-p)^k$.

$\therefore$ probability of getting at least one correct solution is $1 - (1-p)^k$.

Set $k = \ln_{\frac{1}{1-p}}\left(\frac{n^\alpha}{c}\right)$, where $\alpha \geq 1$ and $c > 0$ is a constant.

Then the probability that at least one run produces a correct solution is $1 - (1-p)^k = 1 - \frac{c}{n^\alpha}$.

An event $\Pi$ is said to occur $\underbrace{\text{with high probability}}_{\text{w.h.p.}}$ if $\Pr[\Pi] \geq 1 - \frac{c}{n^\alpha}$.

# Example: A Coloring Problem

Let $S$ be a set of $n$ items.

For $1 \le l \le k$, let $S_l \subseteq S$ such that for every pair of $i, j \in [1, k]$ with $i \ne j$, $S_i \ne S_j$ but not necessarily $S_i \cap S_j = \emptyset$.

For each $l \in [1, k]$, let $|S_l| = r$, where $k \le 2^{r-2}$.

**Problem:** Color each item of $S$ with one of two colors, red and blue, such that each $S_l$ contains at least one red and one blue item.

**Algorithm:** Take each item of $S$ and color it either red or blue independently at random ( with probability $\frac{1}{2}$ ).

Clearly, the algorithm does not always lead to a valid coloring ( i.e., satisfy the constraints given in our problem statement ).

What is the probability that it produces a valid coloring?

# Example: A Coloring Problem

For $1 \le l \le k$, let $R_l$ and $B_l$ be the events that all items of $S_l$ are colored red and blue, respectively.

Then $\Pr[R_l] = \Pr[B_l] = \left(\frac{1}{2}\right)^r = 2^{-r}$ for every $l \in [1, k]$.

$\therefore \Pr\left[\bigcup_{l=1}^{k} R_l\right] = \Pr\left[\bigcup_{l=1}^{k} B_l\right] = k2^{-r} \le 2^{r-2}2^{-r} = \frac{1}{4}.$

Thus $\Pr\left[\bigcup_{l=1}^{k}(R_l \cup B_l)\right] \le 2 \times \frac{1}{4} = \frac{1}{2}.$

$\therefore \Pr\left[\bigcap_{l=1}^{k}(\bar{R}_l \cap \bar{B}_l)\right] = 1 - \Pr\left[\bigcup_{l=1}^{k}(R_l \cup B_l)\right] \ge 1 - \frac{1}{2} = \frac{1}{2}.$

Hence, the algorithm is correct with probability at least $\frac{1}{2}$.

To check if the algorithm has produced a correct result we simply check the items in each $S_l$ to verify that neither $R_l$ nor $B_l$ holds.

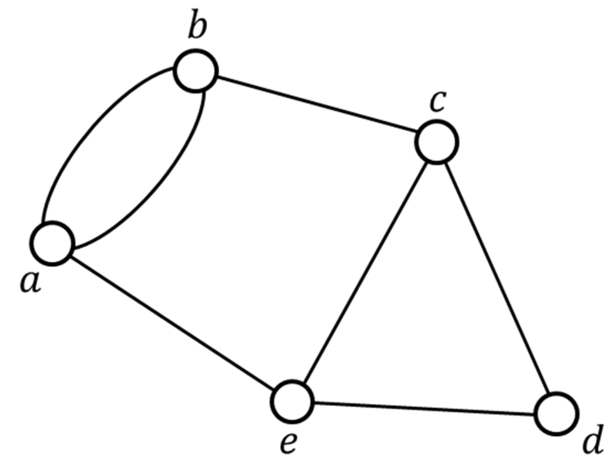
Hence, we can use this simple algorithm to design a Las Vegas algorithm for solving the coloring problem!

# Example: The Min-Cut Problem

Let $G = (V, E)$ be a connected, undirected multigraph with $|V| = n$.

A *cut* in $G$ is a set $C \subseteq E$, such that $G' = (V, E \setminus C)$ is not connected.

A *min-cut* is a cut of minimum cardinality.




The multigraph on the right has a min-cut of size 2: $\{(a, e), (b, c)\}$ and $\{(c, d), (d, e)\}$.

Most deterministic algorithms for finding min-cuts are based on network flows and hence are quite complicated.

Instead in this lecture we will look at a very simple probabilistic algorithm that finds min-cuts with some probability $p > 0$.

# Example: The Min-Cut Problem

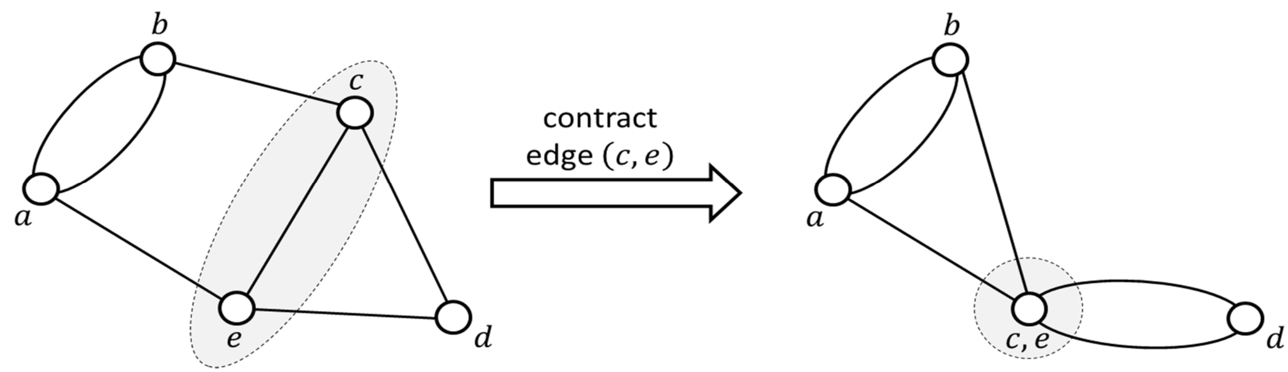
We apply the following contraction step $n - 2$ times on $G = (V, E)$:

Select an edge ( say, $(u, v)$ ) from $E$ uniformly at random.

Merge $u$ and $v$ into a single super vertex.

Remove all edges between $u$ and $v$ from $E$.

If as a result of the contraction there are more than one edges between some pairs of super vertices retain them all.




Let the initial graph be $G_0 = (V_0, E_0)$, where $V_0 = V$ and $E_0 = E$.

Let $G_i = (V_i, E_i)$ be the multigraph after step $i \in [1, n - 2]$.

Then clearly, $|V_i| = n - i$ and thus $|V_{n-2}| = 2$.

We return $E_{n-2}$ as our solution.

# Example: The Min-Cut Problem

Let us fix our attention on a particular min-cut $C$ of $G$.

What is the probability that $E_{n-2} = C$?

Suppose $|C| = k$.

Then each vertex of $G_0 = G$ must have degree at least $k$ as otherwise $G_0$ can be disconnected by removing fewer than $k$ edges.

Hence, $|E| = |E_0| \geq k|V_0|/2 = kn/2$.

Let $\Pi_i$ be the event of not picking an edge of $C$ for contraction in step $i \in [1, n-2]$.

Then clearly, $\Pr[\Pi_1] = 1 - \frac{k}{|E_0|} \geq 1 - \frac{k}{kn/2} = 1 - \frac{2}{n}$

Also $\Pr[\Pi_2|\Pi_1] = 1 - \frac{k}{|E_1|} \geq 1 - \frac{k}{k(n-1)/2} = 1 - \frac{2}{n-1}$

In general, $\Pr\left[\Pi_i | \cap_{j=1}^{i-1} \Pi_j\right] = 1 - \frac{k}{|E_{i-1}|} \geq 1 - \frac{k}{k(n-i+1)/2} = 1 - \frac{2}{n-i+1}$

# Example: The Min-Cut Problem

The probability that no edge of $C$ was ever picked by the algorithm is:

$$\Pr\left[\cap_{i=1}^{n-2} \Pi_i\right] \geq \prod_{i=1}^{n-2}\left(1 - \frac{2}{n-i+1}\right) = \frac{2}{n(n-1)} > \frac{2}{n^2}.$$

So $\Pr[E_{n-2} = C] > \frac{2}{n^2}$, and $\Pr[E_{n-2} \neq C] < 1 - \frac{2}{n^2}$.

Suppose we run the algorithm $n^2/2$ times, and return the smallest cut, say $C'$, obtained from those $n^2/2$ attempts.

Then $\Pr[C' \neq C] < \left(1 - \frac{2}{n^2}\right)^{n^2/2} < \frac{1}{e} \Rightarrow \Pr[C' = C] > 1 - \frac{1}{e}$.

Hence, the algorithm will return a min-cut with probability $> 1 - \frac{1}{e}$.

But we do not know how to detect if the cut returned by the algorithm is, indeed, a min-cut.

Still we can design a Monte-Carlo algorithm based on this simple idea to produce a min-cut with high probability!

# When Only One Success is Not Enough

In both examples we have looked at so far, we were happy with only one success. The analysis was easy.

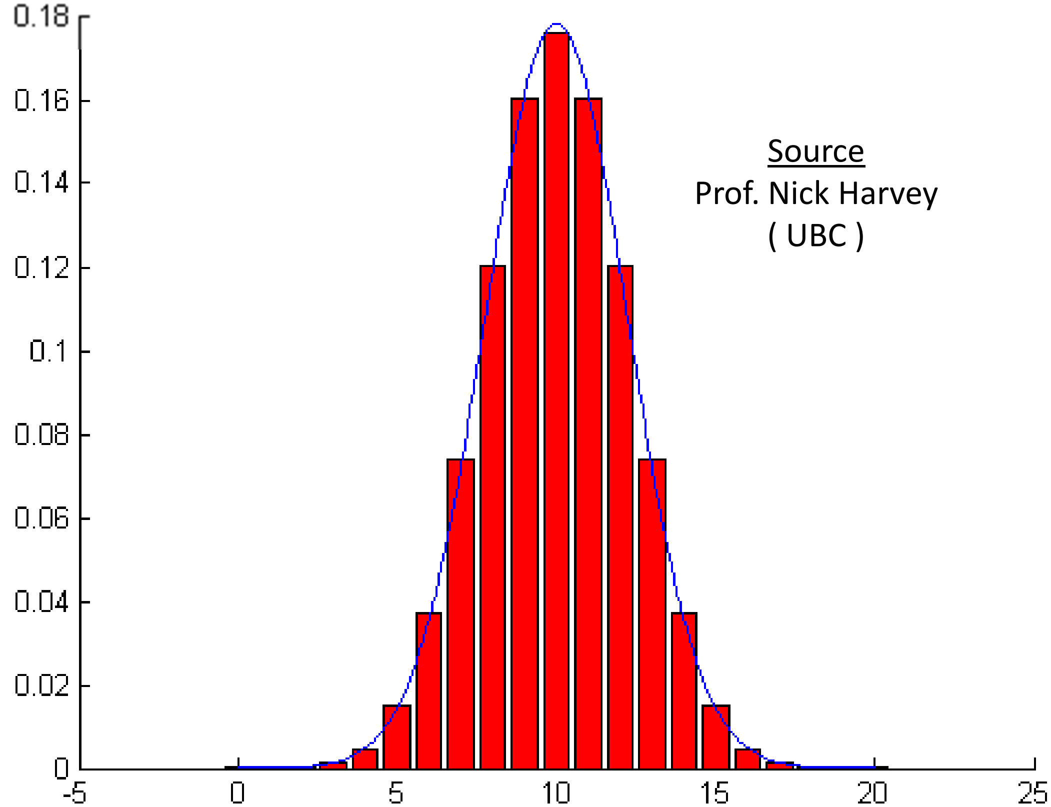
But sometimes we need the algorithm to be successful for at least or at most a certain number of times ( we will see a very familiar such example shortly ).

The number of successful runs required often depends on the size of the input.

How do we analyze those algorithms?

# Binomial Distribution

The binomial distribution is the discrete probability distribution of the #successes in a sequence of $n$ independent yes/no experiments (i.e., Bernoulli trials), each of which succeeds with probability $p$.




RED: Binomial distribution with $n = 20$ and $p = \frac{1}{2}$

Probability mass function:

$$f(k; n, p) = \Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \qquad 0 \le k \le n$$

Cumulative distribution function:

$$F(k; n, p) = \Pr(X \le k) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}, \qquad 0 \le k \le n$$

# Approximating with Normal Distribution

Normal distribution with mean $\mu$ and variance $\sigma^2$ is given by:

$$f(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \qquad x \in \Re$$

For fixed $p$ as $n$ increases the binomial distribution with parameters $n$ and $p$ is well approximated by a normal distribution with $\mu = np$ and $\sigma^2 = np(1-p)$.

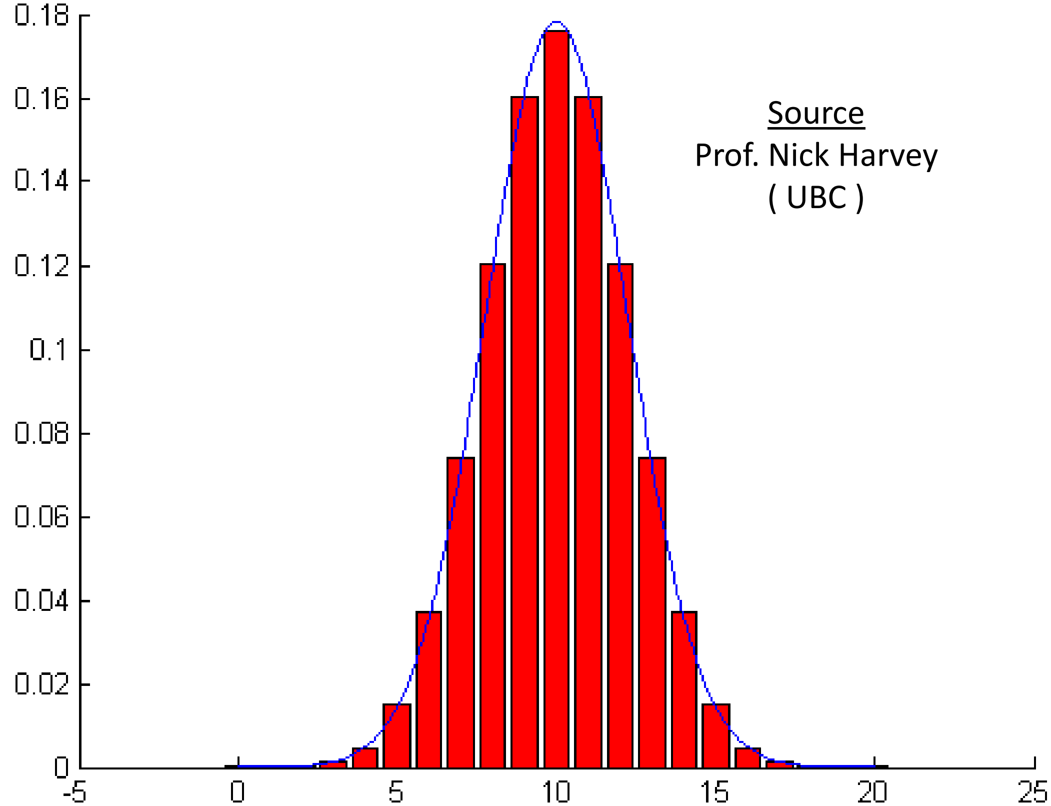
RED: Binomial distribution with

$n = 20$ and $p = \frac{1}{2}$

BLUE: Normal distribution with

$\mu = np = 10$

and $\sigma^2 = np(1-p) = 5$



# Approximating with Normal Distribution

Normal distribution with mean $\mu$ and variance $\sigma^2$ is given by:

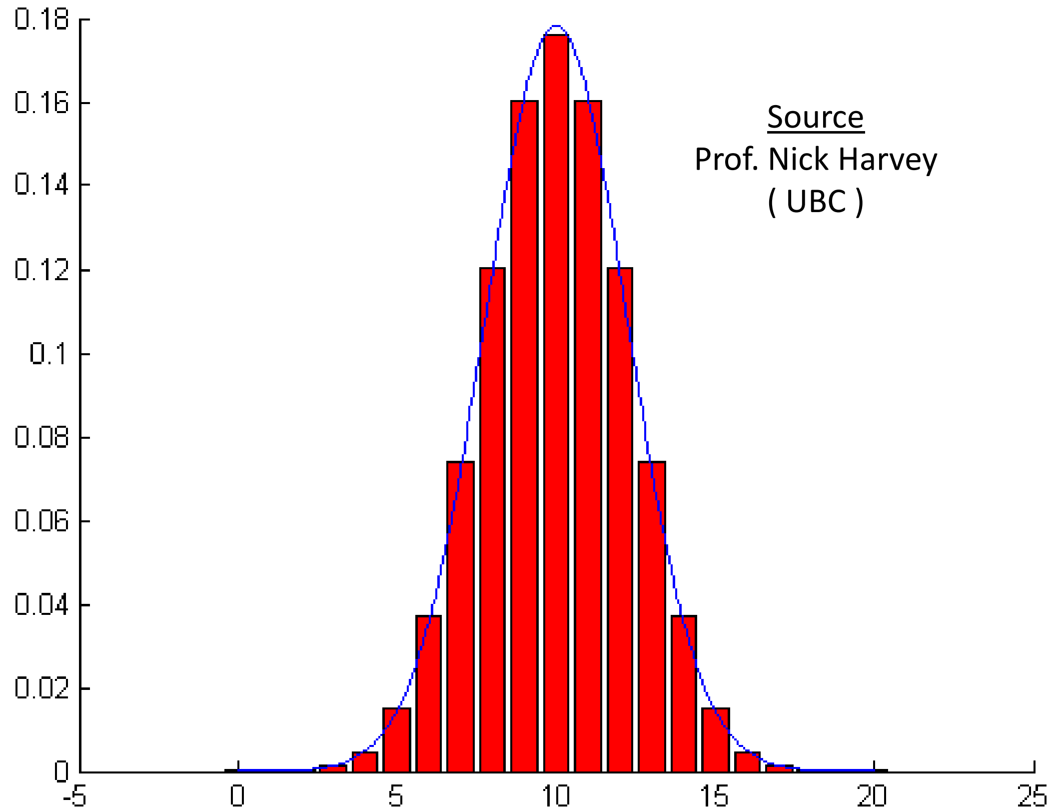
$$f(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \qquad x \in \Re$$

The probability that a normally distributed random variable lies in the interval $(\infty, x]$ is given by:

$$F(x;\mu,\sigma^2) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right],$$

where, $\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$

But $\mathrm{erf}(z)$ cannot be expressed in closed form in terms of elementary functions, and hence difficult to evaluate.



# Approximating with Poisson Distribution

Poisson distribution with mean $\mu > 0$ is given by:

$$f(k;\mu) = \frac{\mu^k e^{-\mu}}{k!}, \qquad k = 0,1,2,\ldots$$

If $np$ is fixed and $n$ increases the binomial distribution with parameters $n$ and $p$ is well approximated by a Poisson distribution with $\mu = np$.

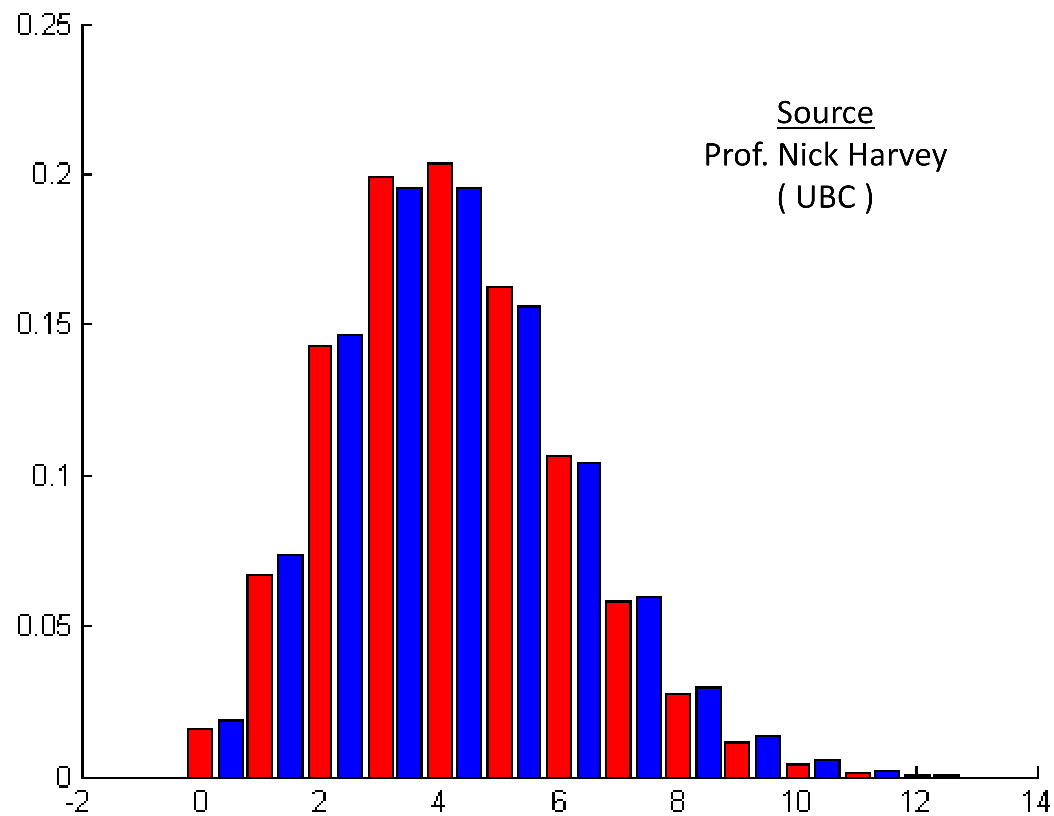
RED: Binomial distribution with

$n = 50$ and $p = \frac{4}{n}$

BLUE: Poisson distribution with

$\mu = np = 200$

Observe that the asymmetry in the plot cannot be well approximated by a symmetric normal distribution.



# Preparing for Chernoff Bounds

**Lemma 1:** Let $X_1, \dots, X_n$ be independent Poisson trials, that is, each $X_i$ is a 0-1 random variable with $\Pr[X_i = 1] = p_i$ for some $p_i$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then for any $t > 0$,

$$E[e^{tX}] \leq e^{(e^t - 1)\mu}.$$

**Proof:** $E[e^{tX_i}] = p_i e^{t\times 1} + (1 - p_i)e^{t\times 0} = p_i e^t + (1 - p_i)$

$$= 1 + p_i(e^t - 1)$$

But for any $y$, $1 + y \leq e^y$. Hence, $E[e^{tX_i}] \leq e^{p_i(e^t - 1)}$.

Now, $E[e^{tX}] = E\left[e^{t\sum_{i=1}^{n} X_i}\right] = E[\prod_{i=1}^{n} e^{tX_i}] = \prod_{i=1}^{n} E[e^{tX_i}]$

$$\leq \prod_{i=1}^{n} e^{p_i(e^t - 1)} = e^{(e^t - 1)\sum_{i=1}^{n} p_i}$$

But, $\mu = E[X] = E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1}^{n} p_i$.

Hence, $E[e^{tX}] \leq e^{(e^t - 1)\mu}$.

# Chernoff Bound 1

**Theorem 3:** Let $X_1, \dots, X_n$ be independent Poisson trials, that is, each $X_i$ is a 0-1 random variable with $\Pr[X_i = 1] = p_i$ for some $p_i$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then for any $\delta > 0$,

$$\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}.$$

**Proof:** Applying Markov's inequality for any $t > 0$,

$$\Pr[X \geq (1+\delta)\mu] = \Pr\left[e^{tX} \geq e^{t(1+\delta)\mu}\right] \leq \frac{E\left[e^{tX}\right]}{e^{t(1+\delta)\mu}}$$

$$\leq \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}} \quad \text{[ Lemma 1 ]}$$

Setting $t = \ln(1+\delta) > 0$, i.e., $e^t = 1+\delta$, we get,

$$\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}.$$

# Chernoff Bound 2

**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem 3, for $\delta > 0$, $\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that for $0 < \delta < 1$, $\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \leq e^{-\frac{\delta^2}{3}}$

$$\Rightarrow \delta - (1+\delta)\ln(1+\delta) \leq -\frac{\delta^2}{3}$$

That is, $f(\delta) = \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$

We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

# Chernoff Bound 2

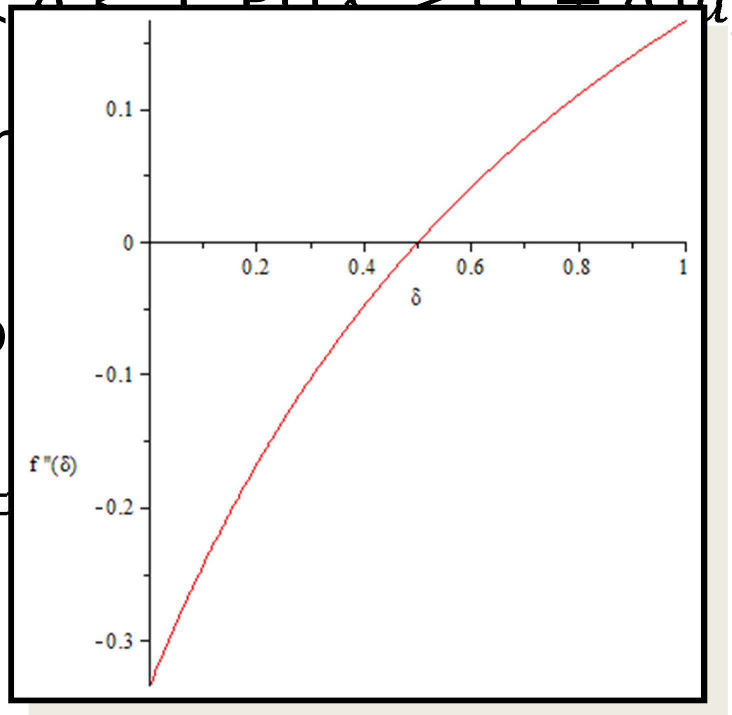
**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem $+ \delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that fo $e^{-\frac{\delta^2}{3}}$

$\leq -\frac{\delta^2}{3}$

That is, $f(\delta) = \delta -$ $0$




We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

# Chernoff Bound 2

**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem 3, for $\delta > 0$, $\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that for $0 < \delta < 1$, $\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \leq e^{-\frac{\delta^2}{3}}$

$$\Rightarrow \delta - (1+\delta)\ln(1+\delta) \leq -\frac{\delta^2}{3}$$

That is, $f(\delta) = \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$

We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

Hence, $f'(\delta)$ first decreases and then increases over $[0,1]$.

Since $f'(0) = 0$ and $f'(1) < 0$, we have $f'(\delta) \leq 0$ over $[0,1]$.

# Chernoff Bound 2

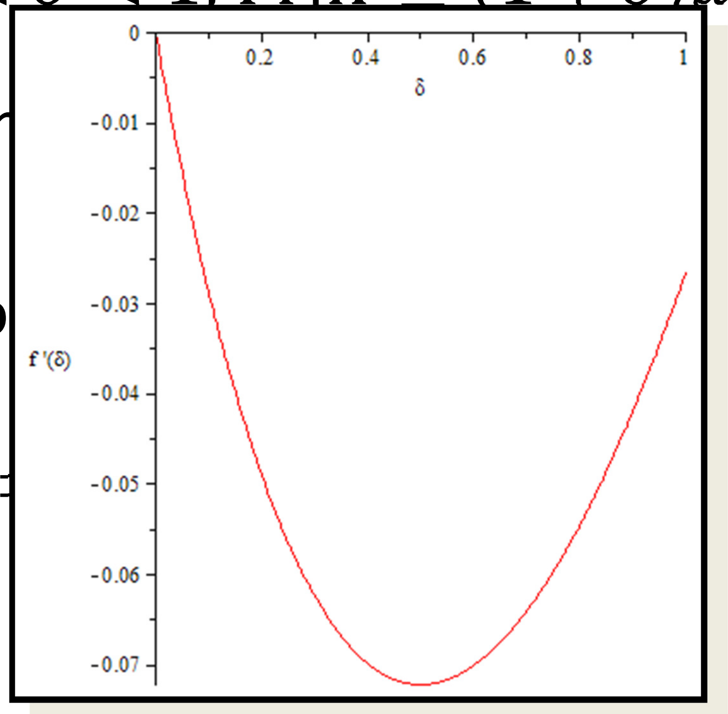
**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem … $+ \delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that fo… $e^{-\frac{\delta^2}{3}}$

… $\leq -\frac{\delta^2}{3}$

That is, $f(\delta) = \delta -$ … $0$




We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

Hence, $f'(\delta)$ first decreases and then increases over $[0,1]$.

Since $f'(0) = 0$ and $f'(1) < 0$, we have $f'(\delta) \leq 0$ over $[0,1]$.

# Chernoff Bound 2

**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem 3, for $\delta > 0$, $\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that for $0 < \delta < 1$, $\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \leq e^{-\frac{\delta^2}{3}}$

$$\Rightarrow \delta - (1+\delta)\ln(1+\delta) \leq -\frac{\delta^2}{3}$$

That is, $f(\delta) = \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$

We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

Hence, $f'(\delta)$ first decreases and then increases over $[0,1]$.

Since $f'(0) = 0$ and $f'(1) < 0$, we have $f'(\delta) \leq 0$ over $[0,1]$.

Since $f(0) = 0$, it follows that $f(\delta) \leq 0$ in that interval.

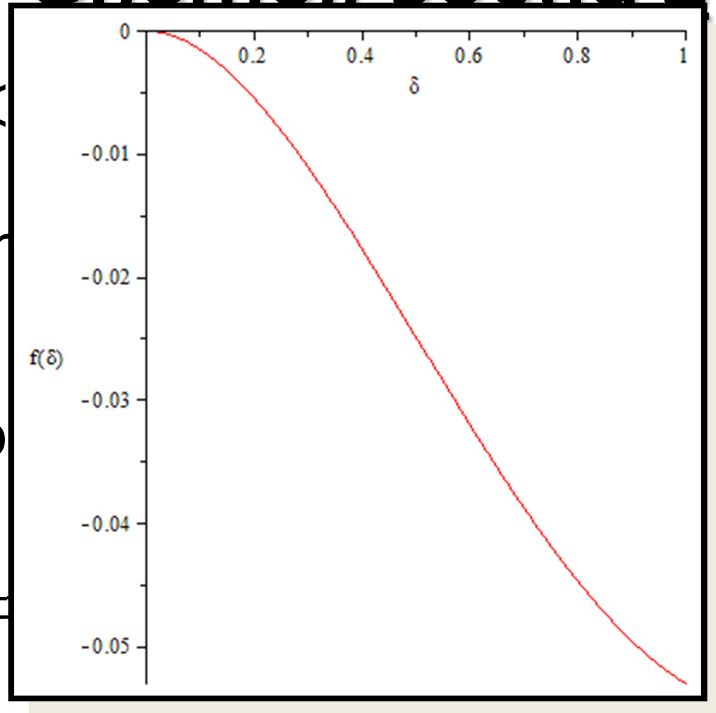
# Chernoff Bound 2

**Theorem 4:** For $0 <$ … $] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem … $+ \delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that fo … $e^{-\frac{\delta^2}{3}}$

… $\leq -\frac{\delta^2}{3}$




That is, $f(\delta) = \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$

We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

Hence, $f'(\delta)$ first decreases and then increases over $[0,1]$.

Since $f'(0) = 0$ and $f'(1) < 0$, we have $f'(\delta) \leq 0$ over $[0,1]$.

Since $f(0) = 0$, it follows that $f(\delta) \leq 0$ in that interval.

# Chernoff Bound 2

**Theorem 4:** For $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\mu\delta^2}{3}}$.

**Proof:** From Theorem 3, for $\delta > 0$, $\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$.

We will show that for $0 < \delta < 1$, $\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \leq e^{-\frac{\delta^2}{3}}$

$$\Rightarrow \delta - (1+\delta)\ln(1+\delta) \leq -\frac{\delta^2}{3}$$

That is, $f(\delta) = \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3} \leq 0$

We have, $f'(\delta) = -\ln(1+\delta) + \frac{2}{3}\delta$, and $f''(\delta) = -\frac{1}{1+\delta} + \frac{2}{3}$

Observe that $f''(\delta) < 0$ for $0 \leq \delta \leq \frac{1}{2}$, and $f''(\delta) > 0$ for $\delta > \frac{1}{2}$.

Hence, $f'(\delta)$ first decreases and then increases over $[0,1]$.

Since $f'(0) = 0$ and $f'(1) < 0$, we have $f'(\delta) \leq 0$ over $[0,1]$.

Since $f(0) = 0$, it follows that $f(\delta) \leq 0$ in that interval.

# Chernoff Bound 3

**Corollary 1:** For $0 < \gamma < \mu$, $\Pr[X \geq \mu + \gamma] \leq e^{-\frac{\gamma^2}{3\mu}}$.

**Proof:** From Theorem 2, for $0 < \delta < 1$, $\Pr[X \geq (1+\delta)\mu] < e^{-\frac{\mu\delta^2}{3}}$.

Setting $\gamma = \mu\delta$, we get, $\Pr[X \geq \mu + \gamma] \leq e^{-\frac{\gamma^2}{3\mu}}$ for $0 < \gamma < \mu$.

# Example: $n$ Fair Coin Flips

$$X_i = \begin{cases} 1 & \text{if the } i\text{th coin flip is heads;} \\ 0 & \text{otherwise.} \end{cases}$$

Then the number of heads in $n$ flips, $X = \sum_{i=1}^{n} X_i$.

We know, $E[X_i] = \Pr[X_i = 1] = \frac{1}{2}$.

Hence, $\mu = E[X] = \sum_{i=1}^{n} E[X_i] = \frac{n}{2}$.

Now putting $\delta = \frac{1}{2}$ in Chernoff bound 2, we have,

$$\Pr\left[X \geq \frac{3n}{4}\right] \leq e^{-\frac{n}{24}} = \frac{1}{e^{\frac{n}{24}}}.$$

# Chernoff Bounds 4, 5 and 6

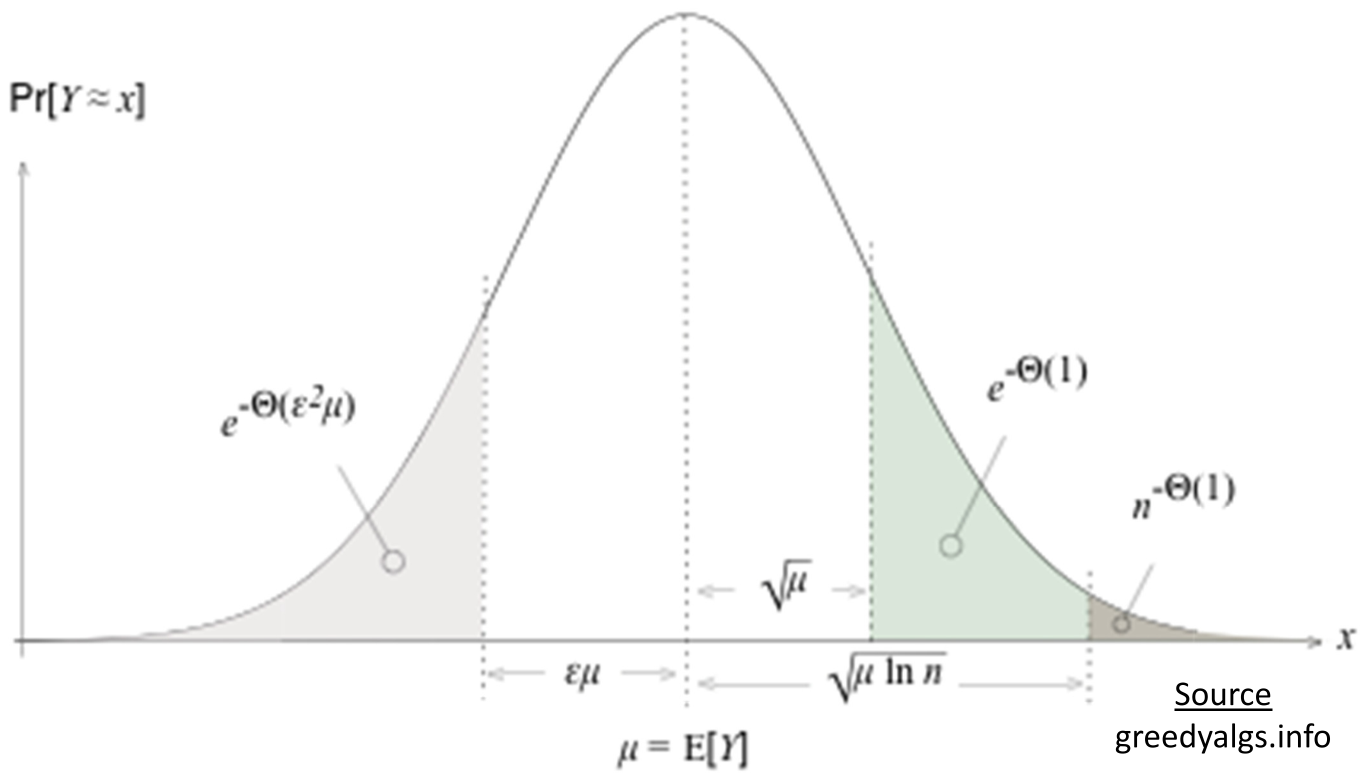
**Theorem 5:** For $0 < \delta < 1$, $\Pr[X \leq (1-\delta)\mu] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}$.

**Theorem 6:** For $0 < \delta < 1$, $\Pr[X \leq (1-\delta)\mu] \leq e^{-\frac{\mu\delta^2}{2}}$.

**Corollary 2:** For $0 < \gamma < \mu$, $\Pr[X \leq \mu - \gamma] \leq e^{-\frac{\gamma^2}{2\mu}}$.

# Chernoff Bounds

| Lower Tail | Upper Tail |
|---|---|
| $\mathbf{0} < \boldsymbol{\delta} < \mathbf{1}$: $\Pr[X \le (1-\delta)\mu] \le \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}$ | $\boldsymbol{\delta} > \mathbf{0}$: $\Pr[X \ge (1+\delta)\mu] \le \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$ |
| $\mathbf{0} < \boldsymbol{\delta} < \mathbf{1}$: $\Pr[X \le (1-\delta)\mu] \le e^{-\frac{\mu\delta^2}{2}}$ | $\mathbf{0} < \boldsymbol{\delta} < \mathbf{1}$: $\Pr[X \ge (1+\delta)\mu] \le e^{-\frac{\mu\delta^2}{3}}$ |
| $\mathbf{0} < \boldsymbol{\gamma} < \boldsymbol{\mu}$: $\Pr[X \le \mu - \gamma] \le e^{-\frac{\gamma^2}{2\mu}}$ | $\mathbf{0} < \boldsymbol{\gamma} < \boldsymbol{\mu}$: $\Pr[X \ge \mu + \gamma] \le e^{-\frac{\gamma^2}{3\mu}}$ |




Source
greedyalgs.info

# Randomized Quicksort
# ( RANDQS )

# Randomized Quicksort ( RANDQS )

**Input:** A set of numbers $S$. ( i.e., all numbers are distinct )

**Output:** The numbers of $S$ sorted in increasing order.

**Steps:**

1. **Pivot Selection:** Select a number $x \in S$ uniformly at random.
2. **Partition:** Compare each number of $S$ with $x$, and determine sets $S_l = \{y \in S \mid y < x\}$ and $S_r = \{y \in S \mid y > x\}$.
3. **Recursion:** Recursively sort $S_l$ and $S_r$.
4. **Output:** Output the sorted version of $S_l$, followed by $x$, followed by the sorted version of $S_r$.

# Randomized Quicksort ( RANDQS )

**Input:** A set of numbers $S$. ( i.e., all numbers are distinct )

**Output:** The numbers of $S$ sorted in increasing order.

**Steps:**

1. **Pivot Selection:** Select a number $x \in S$ uniformly at random.
2. **Partition:** Compare each number of $S$ with $x$, and determine sets $S_l = \{y \in S |\ y < x\}$ and $S_r = \{y \in S |\ y > x\}$.
3. **Recursion:** Recursively sort $S_l$ and $S_r$.
4. **Output:** Output the sorted version of $S_l$, followed by $x$, followed by the sorted version of $S_r$.

**Assumption:** RANDQS is called only on nonempty $S$.

**Observation:** If $|S| = n$, fewer than $n$ recursive calls to RANDQS will be made during the sorting of $S$. ( why? )

**Observation:** If $|S| = n$, and $X$ is the total number of comparisons made in step 2 ( Partition ) across all ( original and recursive ) calls to RANDQS, then RANDQS sorts $S$ in $\mathrm{O}(n + X)$ time.

# Expected Running Time of RANDQS

**Observation:** If $|S| = n$, and $X$ is the total number of comparisons made in step 2 ( Partition ) across all ( original and recursive ) calls to RANDQS, then RANDQS sorts $S$ in $\mathrm{O}(n + X)$ time.

Then all we need to do is determine $E[X]$.

Let $s_1, s_2, \dots, s_n$ be the elements of $S$ in sorted order.

Let $S_{ij} = \{s_i, s_{i+1}, \dots, s_j\}$ for all $1 \le i < j \le n$.

Observe that each pair of elements of $S$ is compared at most once during the entire execution of the algorithm. ( why? )

For $1 \le i < j \le n$, let $X_{ij} = \begin{cases} 1 & \text{if } s_i \text{ is compared to } s_j; \\ 0 & \text{otherwise.} \end{cases}$

Then $X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$ .

# Expected Running Time of RANDQS

For $1 \le i < j \le n$, let $X_{ij} = \begin{cases} 1 & \text{if } s_i \text{ is compared to } s_j; \\ 0 & \text{otherwise.} \end{cases}$

Then $X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$

$$\Rightarrow E[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} E\left[X_{ij}\right]$$
$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \Pr\left[X_{ij} = 1\right]$$

**Observations:**

$\boldsymbol{X_{ij} = 0}$**:** Once a pivot $x$ with $s_i < x < s_j$ is chosen, $s_i$ and $s_j$ will never be compared at any subsequent time. ( why? )

$\boldsymbol{X_{ij} = 1}$**:** If either $s_i$ or $s_j$ is chosen as a pivot before any other item in $S_{ij}$ then $s_i$ will be compared with $s_j$. ( why? )

# Expected Running Time of RANDQS

Since each element of $S_{ij}$ is equally likely to be chosen as a pivot:

$$\Pr[X_{ij} = 1] \leq \frac{1}{j-i+1} + \frac{1}{j-i+1} = \frac{2}{j-i+1}.$$

Hence, $E[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \Pr[X_{ij} = 1]$

$$\leq \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{2}{j-i+1}$$

$$= \sum_{i=1}^{n-1} \sum_{k=1}^{n-i} \frac{2}{k+1}$$

$$< \sum_{i=1}^{n-1} \sum_{k=1}^{n} \frac{2}{k}$$

$$= \sum_{i=1}^{n-1} \mathrm{O}(\log n)$$

$$= \mathrm{O}(n \log n)$$

Thus expected running time of RANDQS is $\mathrm{O}(n \log n)$.

# High Probability Bound for RANDQS

**Input:** A set of numbers $S$. ( i.e., all numbers are distinct )

**Output:** The numbers of $S$ sorted in increasing order.

**Steps:**

1. **Pivot Selection:** Select a number $x \in S$ uniformly at random.
2. **Partition:** Compare each number of $S$ with $x$, and determine sets $S_l = \{y \in S \mid y < x\}$ and $S_r = \{y \in S \mid y > x\}$.
3. **Recursion:** Recursively sort $S_l$ and $S_r$.
4. **Output:** Output the sorted version of $S_l$, followed by $x$, followed by the sorted version of $S_r$.

We will prove that w.h.p. the running time of RANDQS does not exceed its expected running time by more than a constant factor.

In other words, we show that w.h.p. RANDQS runs in $\mathrm{O}(n \log n)$ time.

# High Probability Bound for RANDQS

**Input:** A set of numbers $S$. ( i.e., all numbers are distinct )

**Output:** The numbers of $S$ sorted in increasing order.

**Steps:**

1. **Pivot Selection:** Select a number $x \in S$ uniformly at random.
2. **Partition:** Compare each number of $S$ with $x$, and determine sets $S_l = \{y \in S |\ y < x\}$ and $S_r = \{y \in S |\ y > x\}$.
3. **Recursion:** Recursively sort $S_l$ and $S_r$.
4. **Output:** Output the sorted version of $S_l$, followed by $x$, followed by the sorted version of $S_r$.

Let us fix an element $z$ in the original input set of size $n$.

We will trace the partition containing $z$ for $c \ln n$ levels of recursion, where $c$ is a constant to be determined later.

If a partitioning step divides $S$ such that $\frac{|S|}{4} \leq |S_l|, |S_r| \leq \frac{3|S|}{4}$, we call that partition a *balanced* partition.

# High Probability Bound for RANDQS

We will prove that among the $c \ln n$ partitioning steps $z$ undergoes, w.h.p. at least $\frac{c}{4} \ln n$ results in balanced partitions.

If at any point $z$ is in a partition of size $k$, after a balanced partitioning step it ends up in a partition of size at most $\left(\frac{3}{4}\right) k$.

Since the input size is $n$, after $\frac{c}{4} \ln n$ balanced partitions, $z$ will end up in a partition of size $\leq \left(\frac{3}{4}\right)^{\frac{c}{4} \ln n} n = \frac{n}{n^{\frac{c}{4} \ln\left(\frac{4}{3}\right)}}$, which is $\leq 1$ for $c \geq 14$.

That means if $c \geq 14$, then $z$ will end up in its final sorted position in the output after undergoing $\frac{c}{4} \ln n$ balanced partitions.

# High Probability Bound for RANDQS

For $1 \leq i \leq c \ln n$, let

$$Z_i = \begin{cases} 1 & \text{if the partition at recursion level } i \text{ is balanced;} \\ 0 & \text{otherwise.} \end{cases}$$

But a balanced partition is obtained by choosing a pivot with rank between $\frac{k}{4}$ and $\frac{3k}{4}$, where $k$ is the size of the set being partitioned.

Since each element of the set is chosen as a pivot uniformly at random, a balancing pivot will be chosen with probability $\frac{\frac{3k}{4}-\frac{k}{4}}{k} = \frac{1}{2}$.

Hence, $\Pr[Z_i = 1] = \frac{1}{2}$.

Thus $E[Z_i] = \Pr[Z_i = 1] = \frac{1}{2}$.

# High Probability Bound for RANDQS

Total number of balanced partitions, $Z = \sum_{i=1}^{c \ln n} Z_i$.

Then $\mu = E[Z] = \sum_{i=1}^{c \ln n} E[Z_i] = \frac{c \ln n}{2}$.

Now applying Chernoff bound 5 ( see Theorem 6 ) with $\delta = \frac{1}{2}$,

$$\Pr[Z \le (1 - \delta)\mu] \le e^{-\frac{\mu\delta^2}{2}}$$

$$\Rightarrow \Pr\left[Z \le \frac{c}{4} \ln n\right] \le e^{-\frac{\mu\delta^2}{2}} = e^{-\frac{c}{16} \ln n} = n^{-\frac{c}{16}} = \frac{1}{n^{\frac{c}{16}}}.$$

For $c = 32$, we have $\Pr[Z \le 8 \ln n] \le \frac{1}{n^2}$.

This means that the probability that $z$ fails to reach its final sorted

position even after $32 \ln n$ levels of recursion is $\le \frac{1}{n^2}$.

# High Probability Bound for RANDQS

The probability that at least one of $n$ input elements fails to reach its final sorted position after $32 \ln n$ levels of recursion is $\leq n \times \frac{1}{n^2} = \frac{1}{n}$.

$\therefore$ the probability that all $n$ input elements reach their final sorted positions after $32 \ln n$ levels of recursion is $\geq 1 - \frac{1}{n}$.

But observe that the total amount of work done in each level of recursion is $\mathrm{O}(n)$.

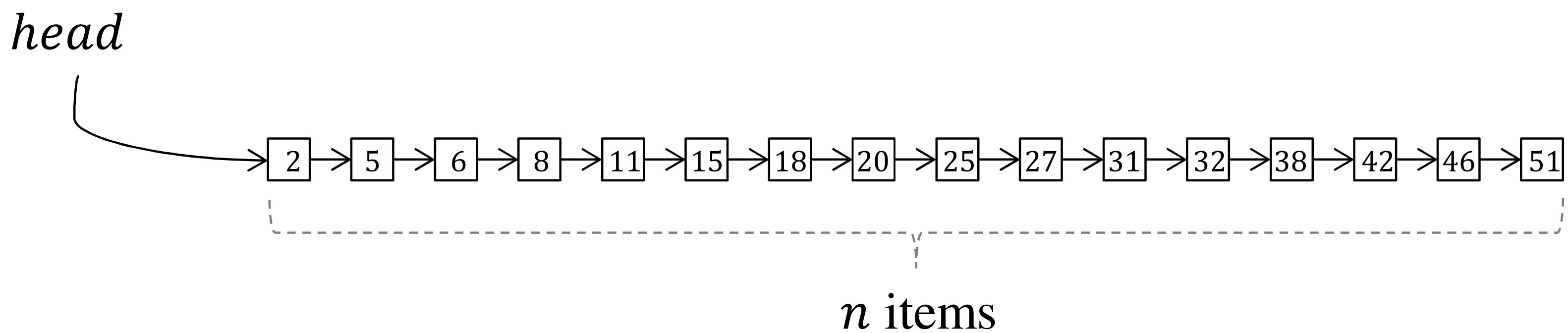
$\therefore$ total work done in $32 \ln n$ levels of recursion is $\mathrm{O}(n \log n)$.

Hence, w.h.p. RANDQS terminates in $\mathrm{O}(n \log n)$ time.

# Random Skip Lists

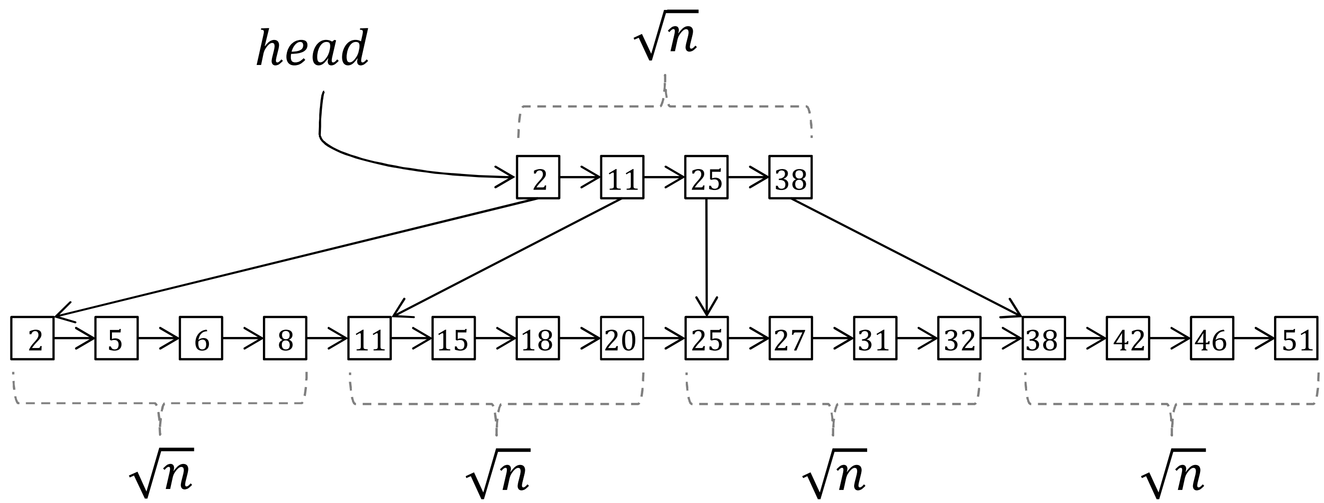
# Searching in a Sorted Linked List

## Traditional Linked List:




SEARCH( $x$ ): Takes $\leq n$ time.

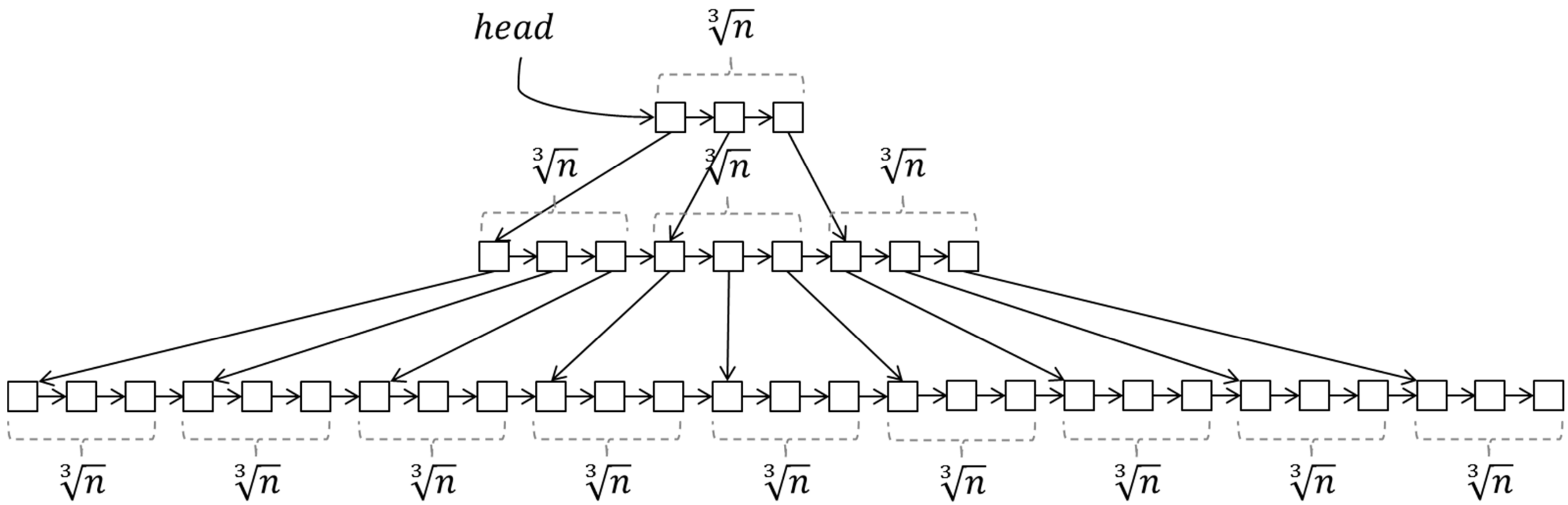
## 2-level Linked List:




SEARCH( $x$ ): Takes $\leq 2\sqrt{n}$ time.

# Searching in a Sorted Linked List

**3-level Linked List:**



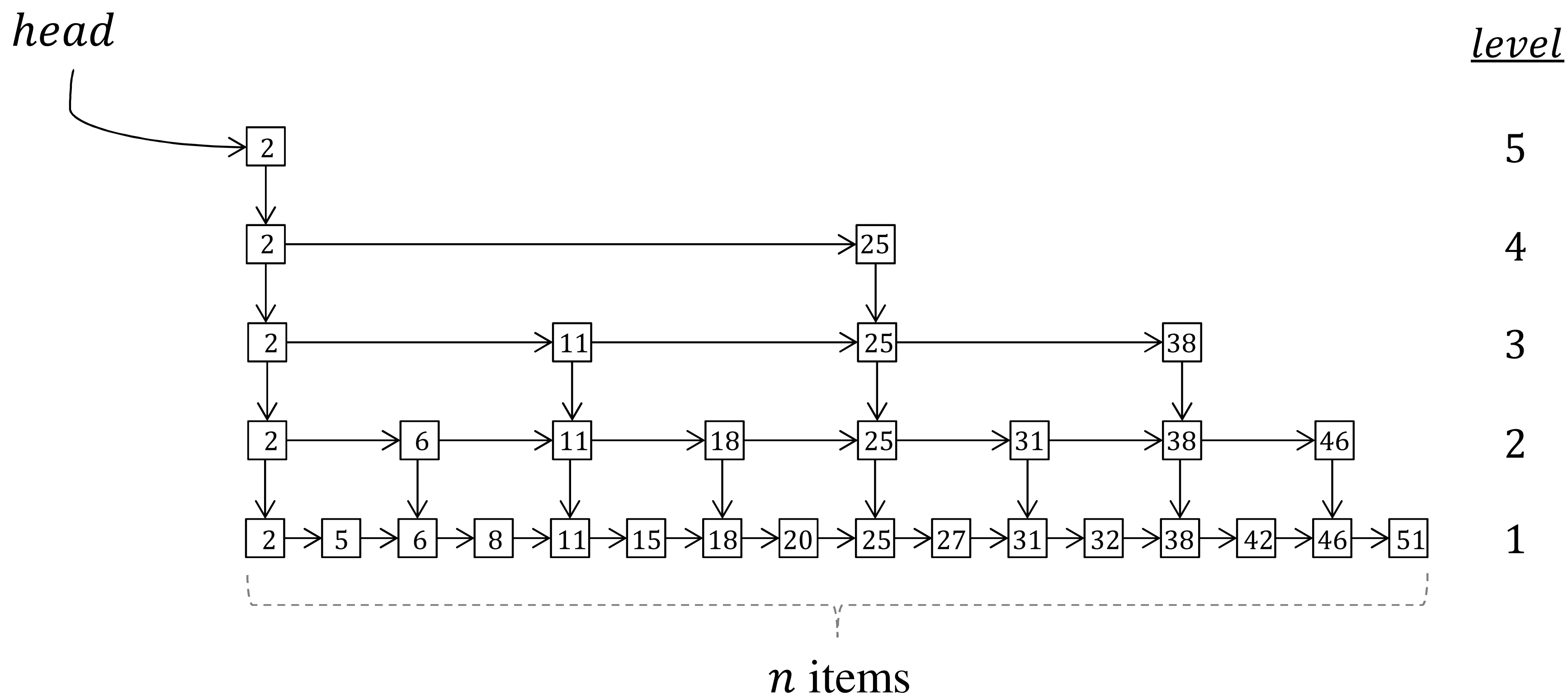

SEARCH( $x$ ): Takes $\leq 3\sqrt[3]{n}$ time.

***k*-level Linked List:** SEARCH( $x$ ) takes $\leq k\sqrt[k]{n} = kn^{\frac{1}{k}}$ time.

**For $\boldsymbol{k = \log n}$:** SEARCH( $x$ ) takes $\leq (\log n) \cdot n^{\frac{1}{\log n}} = 2\log n$ time!

# Searching in a Sorted Linked List

**$(\log n)$-level Linked List:** SEARCH takes $\leq (\log n) \cdot n^{\frac{1}{\log n}} = 2 \log n$ time!

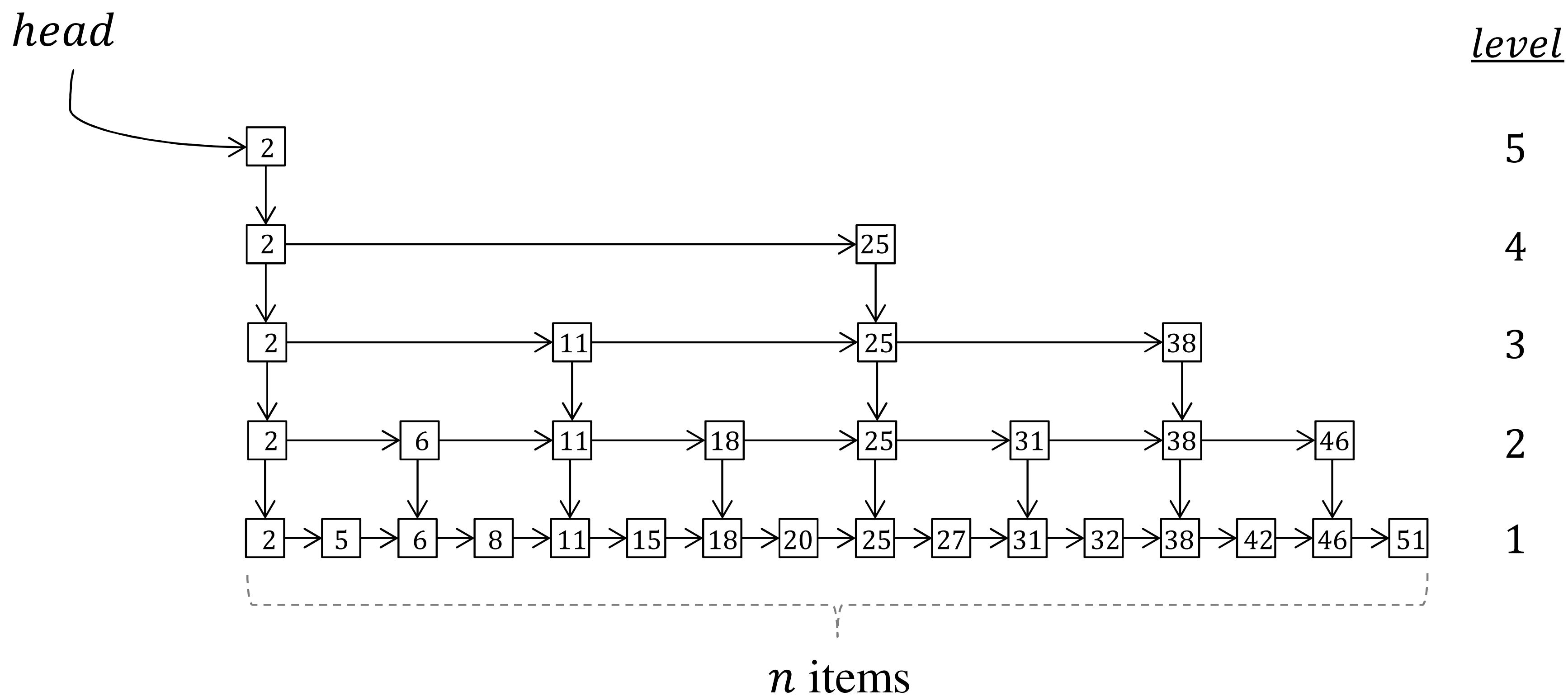



**Observations:**

1. Let $n_l$ = #items in level $l$. Then $n_{l+1} = \left\lceil \frac{n_l}{2} \right\rceil$.
2. Let $m_l = n_l - n_{l+1}$ = #items in level $l$ that have not reached level $l + 1$. Then $m_l = \left\lfloor \frac{n}{2^l} \right\rfloor$.

# Searching in a Sorted Linked List

$(\mathbf{log}\, \boldsymbol{n})$**-level Linked List:** SEARCH takes $\leq (\log n) \cdot n^{\frac{1}{\log n}} = 2 \log n$ time!

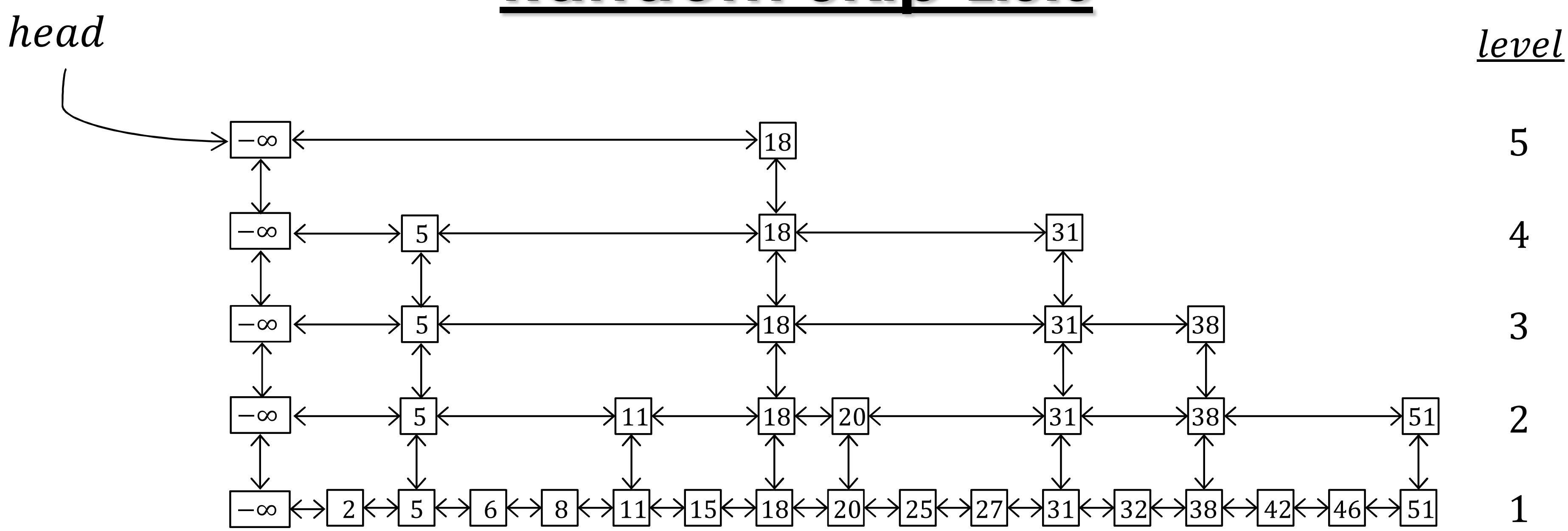



How do we maintain this regular structure under insertion and deletion of items?

Deterministic solution does not seem straightforward.

But randomization can make life really easy!

# Random Skip Lists

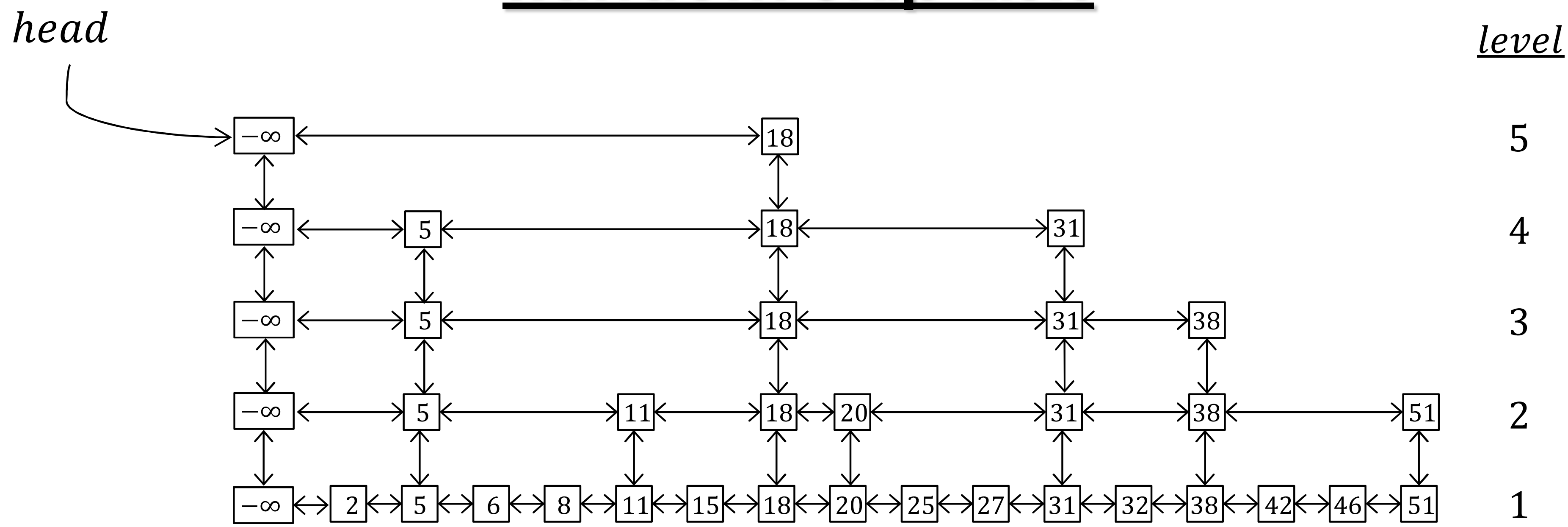



**Construction:**

1. Start with all items along with a sentinel $-\infty$ in level 1.
2. Promote each non-sentinel item of level $l > 0$ to level $l+1$ with probability $\frac{1}{2}$.

   If level $l+1$ is nonempty promote the sentinel, too.

# Random Skip Lists



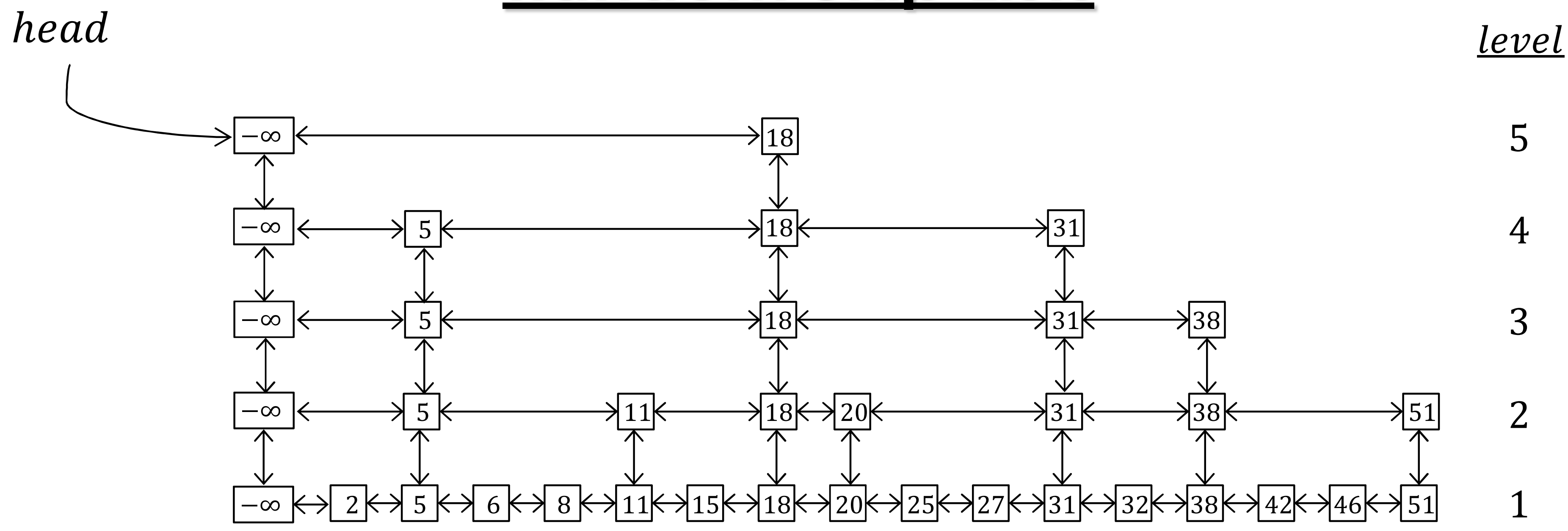

Let $L$ be a skip list,

$L_k$ be the set of all items in level $k \geq 1$,

$l(x) = \max\{\, k \mid x \in L_k\}$, and

$h(L) = \max\{\, l(x) \mid x \in L_0 \,\}$.

# Random Skip Lists




Clearly, for each $x \in L$ and $k \geq 1$, $\Pr[\ l(x) = k\ ] = \frac{1}{2^k}$ .

Then $\Pr[\ l(x) > k\ ] = \sum_{i=k+1}^{\infty} \Pr[\ l(x) = i\ ] = \sum_{i=k+1}^{\infty} \left(\frac{1}{2^i}\right) = \frac{1}{2^k}$ .

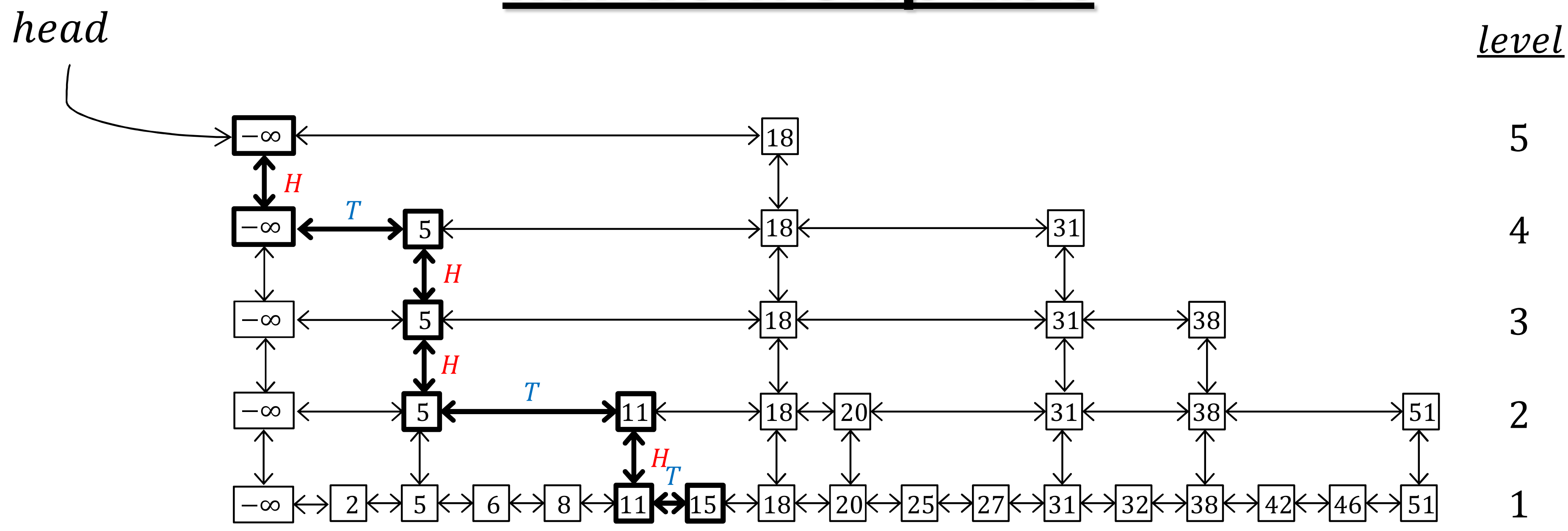
$\therefore \Pr[\ h(L) > k\ ] = \sum_{x \in L} \Pr[\ l(x) > k\ ] = \frac{n}{2^k}$ .

$\Rightarrow \Pr[\ h(L) \leq k\ ] = 1 - \Pr[\ h(L) > k\ ] = 1 - \frac{n}{2^k}$ .

$\therefore$ For constant $c > 2$, $\Pr[\ h(L) \leq c \log n\ ] = 1 - \frac{n}{2^{c \log n}} = 1 - \frac{1}{n^{c-1}}$ .

Hence, w.h.p. height of a skip list is $\mathrm{O}(\log n)$.

# Random Skip Lists




Let us flip $4c\log n$ fair coins, and let $X$ is the number of heads we get.

Then $\mu = E[X] = (4c\log n) \times \frac{1}{2} = 2c\log n$ .

We know for $0 < \delta < 1$, Chernoff bound, $\Pr[X \le (1-\delta)\mu] \le e^{-\frac{\mu\delta^2}{2}}$.

Putting $\delta = \frac{1}{2}$ and $\mu = 2c\log n$, we get, $\Pr[X \le c\log n\,] \le \frac{1}{n^{\frac{c}{4}}}$ .

For $c \ge 16$, $\Pr[X > c\log n] \ge 1 - \frac{1}{n}$ .

Hence, w.h.p. we will get more than $c\log n$ heads.