A random variable $X$ on a sample space $\Omega$ is a real-valued function on $\Omega$; that is, $X: \Omega \rightarrow \mathbb{R}$. A discrete random variable is a random variable that takes on only a finite or countably infinite number of values.

Discrete random variable $X$ and real value $a$: the event “$X = a$” represents the set $\{s \in \Omega : X(s) = a\}$.

$$
\Pr(X = a) = \sum_{s \in \Omega: X(s) = a} \Pr(s)
$$
**Independence**

**Definition**

Two random variables $X$ and $Y$ are independent if and only if

$$\Pr((X = x) \cap (Y = y)) = \Pr(X = x) \cdot \Pr(Y = y)$$

for all values $x$ and $y$. Similarly, random variables $X_1, X_2, \ldots, X_k$ are mutually independent if and only if for any subset $I \subseteq [1, k]$ and any values $x_i, i \in I$,

$$\Pr\left(\bigcap_{i \in I} X_i = x_i\right) = \prod_{i \in I} \Pr(X_i = x_i).$$
Expectation

**Definition**

The *expectation* of a discrete random variable $X$, denoted by $\mathbb{E}[X]$, is given by

$$
\mathbb{E}[X] = \sum_i i \Pr(X = i),
$$

where the summation is over all values in the range of $X$. The expectation is finite if $\sum_i |i| \Pr(X = i)$ converges; otherwise, the expectation is unbounded.

The expectation (or mean or average) is a weighted sum over all possible values of the random variable.
Definition

The **median** of a random variable $X$ is a value $m$ such

$$Pr(X < m) \leq 1/2 \quad \text{and} \quad Pr(X > m) < 1/2.$$
Linearity of Expectation

Theorem

For any two random variables $X$ and $Y$

$$E[X + Y] = E[X] + E[Y].$$

Lemma

For any constant $c$ and discrete random variable $X$,

$$E[cX] = cE[X].$$
A Bernoulli or an indicator random variable:

\[ Y = \begin{cases} 
1 & \text{if the experiment succeeds,} \\
0 & \text{otherwise.} 
\end{cases} \]

Let \( \Pr(Y = 1) = p \).

\[ \mathbb{E}[Y] = p \cdot 1 + (1 - p) \cdot 0 = p = \Pr(Y = 1). \]
A binomial random variable $X$ with parameters $n$ and $p$, denoted by $B(n, p)$, is defined by the following probability distribution on $j = 0, 1, 2, \ldots, n$:

$$
\Pr(X = j) = \binom{n}{j} p^j (1 - p)^{n-j}.
$$
Expectation of a Binomial Random Variable

\[ E[X] = \sum_{j=0}^{n} j \left( \begin{array}{c} n \\ j \end{array} \right) p^j (1 - p)^{n-j} \]

\[ = \sum_{j=0}^{n} j \frac{n!}{j!(n-j)!} p^j (1 - p)^{n-j} \]

\[ = \sum_{j=1}^{n} \frac{n!}{(j - 1)!(n-j)!} p^j (1 - p)^{n-j} \]

\[ = np \sum_{j=1}^{n} \frac{(n-1)!}{(j-1)!((n-1)-(j-1))!} p^{j-1}(1 - p)^{(n-1)-(j-1)} \]

\[ = np \sum_{k=0}^{n-1} \frac{(n-1)!}{k!((n-1)-k)!} p^k (1 - p)^{(n-1)-k} \]

\[ = np \sum_{k=0}^{n-1} \left( \begin{array}{c} n-1 \\ k \end{array} \right) p^k (1 - p)^{(n-1)-k} = np. \]
Binomial Random Variable

$X$ represents the number of successes in $n$ independent experiments, each with success probability $p$. Let

$$X_i = \begin{cases} 
1 & \text{if the } i\text{-th experiment succeeds,} \\
0 & \text{otherwise.}
\end{cases}$$

Using linearity of expectations

$$E[X] = E \left[ \sum_{i=1}^{n} X_i \right] = \sum_{i=1}^{n} E[X_i] = np.$$
Example: Finding the \( k \)-Smallest Element

Procedure \text{Order}(S, k);

\textbf{Input:} A set \( S \), an integer \( k \leq |S| = n \).

\textbf{Output:} The \( k \) smallest element in the set \( S \).
Example: Finding the $k$-Smallest Element

Procedure $\text{Order}(S, k)$;

**Input:** A set $S$, an integer $k \leq |S| = n$.

**Output:** The $k$ smallest element in the set $S$.

1. If $|S| = k = 1$ return $S$.
2. Choose a random element $y$ uniformly from $S$.
3. Compare all elements of $S$ to $y$. Let $S_1 = \{x \in S \mid x \leq y\}$ and $S_2 = \{x \in S \mid x > y\}$.
4. If $k \leq |S_1|$ return $\text{Order}(S_1, k)$ else return $\text{Order}(S_2, k - |S_1|)$.

**Theorem**

The algorithm returns the $k$-smallest element in $S$ and performs $O(n)$ comparisons in expectation.
Proof

• We say that a call to Order\((S, k)\) was *successful* if the random element was in the middle \(1/3\) of the set \(S\). A call is successful with probability \(1/3\).
• After the \(i\)-th successful call the size of the set \(S\) is bounded by \(n(2/3)^i\). Thus, need at most \(\log_{3/2} n\) successful calls.
• Let \(X\) be the total number of comparisons. Let \(X_i\) be the number of comparisons between the \(i\)-th successful call (included) and the \(i + 1\)-th (excluded): \(\mathbb{E}[X] \leq \sum_{i=0}^{\log_{3/2} n} \mathbb{E}[X_i]\).
• Let \(Y_i\) \((i > 0)\) be the number of calls between the \(i\)-th successful call and the \(i + 1\)-th successful call. The expected number of calls from one successful call to the next successful call (included) is 3: \(\mathbb{E}[Y_i] = 3\) for all \(i\).
• Therefore \(\mathbb{E}[X_i] \leq n(2/3)^i \mathbb{E}[Y_i] = 3n(2/3)^i\).
• Expected number of comparisons:

\[
\mathbb{E}[X] \leq \sum_{j=0}^{\log_{3/2} n} 3n \left(\frac{2}{3}\right)^j \leq 9n.
\]
Definition

A geometric random variable $X$ with parameter $p$ is given by the following probability distribution on $n = 1, 2, \ldots$.

$$\Pr(X = n) = (1 - p)^{n-1} p.$$ 

Example: repeatedly draw independent Bernoulli random variables with parameter $p > 0$ until we get a 1. Let $X$ be number of trials up to and including the first 1. Then $X$ is a geometric random variable with parameter $p$. 
Memoryless Property

Lemma

For a geometric random variable with parameter $p$ and $n > 0$,

$$\Pr(X = n + k \mid X > k) = \Pr(X = n).$$

Proof.

$$\Pr(X = n + k \mid X > k) = \frac{\Pr((X = n + k) \cap (X > k))}{\Pr(X > k)}$$

$$= \frac{\Pr(X = n + k)}{\Pr(X > k)}$$

$$= \frac{(1 - p)^{n+k-1}p}{\sum_{i=k}^{\infty}(1 - p)^i p}$$

$$= \frac{(1 - p)^{n+k-1}p}{(1 - p)^k}$$

$$= (1 - p)^{n-1}p$$

$$= \Pr(X = n).$$
I have a coin and flip it a number of times \textbf{without telling you neither the outcomes nor the number of times I flip it.}

I give the coin to you, asking \textbf{what is the probability that the first head will come out after} \( n \) \textbf{times you have flipped it.}

Intuitively, this probability \textbf{should not depend on the number of times I flip the coin} and/or on the corresponding outcomes.

The memoryless property tells us it is indeed so.
Lemma

Let $X$ be a discrete random variable that takes on only non-negative integer values. Then

$$\mathbb{E}[X] = \sum_{i=1}^{\infty} \Pr(X \geq i).$$

Proof.

$$\sum_{i=1}^{\infty} \Pr(X \geq i) = \sum_{i=1}^{\infty} \sum_{j=i}^{\infty} \Pr(X = j)$$

$$= \sum_{j=1}^{\infty} \sum_{i=1}^{j} \Pr(X = j)$$

$$= \sum_{j=1}^{\infty} j \Pr(X = j) = \mathbb{E}[X].$$
For a geometric random variable $X$ with parameter $p$,

$$
\Pr(X \geq i) = \sum_{n=i}^{\infty} (1 - p)^{n-1} p = (1 - p)^{i-1}.
$$

$$
\mathbb{E}[X] = \sum_{i=1}^{\infty} \Pr(X \geq i) = \sum_{i=1}^{\infty} (1 - p)^{i-1} = \frac{1}{1 - (1 - p)} = \frac{1}{p}
$$
Alternative Proof

For a geometric random variable $X$ with parameter $p$,

$$\Pr(X \geq i) = \sum_{n=i}^{\infty} (1 - p)^{n-1} p = (1 - p)^{i-1}.$$ 

Let $E_1$ be the event "success in the first trial"

$$E[X] = E[X | E_1]Pr(E_1) + E[X | \bar{E}_1]Pr(\bar{E}_1)$$

$$= 1 \cdot p + (1 + E[X])(1 - p)$$

$$= 1 + (1 - p)E[X]$$

Thus,

$$E[X] = \frac{1}{p}.$$
Example: Finding the $k$-Smallest Element

Procedure Order($S, k$);

**Input:** An array $S$, an integer $k \leq |S| = n$.

**Output:** The $k$ smallest element in the set $S$.

1. If $|S| = k = 1$ return $S$.
2. Let $y$ be the first element is $S$.
3. Compare all elements of $S$ to $y$. Let $S_1 = \{x \in S \mid x \leq y\}$ and $S_2 = \{x \in S \mid x > y\}$.
4. If $k \leq |S_1|$ return Order($S_1, k$) else return Order($S_2, k - |S_1|$).

**Theorem**

The algorithm returns the $k$-smallest element in $S$ and performs $O(n)$ comparisons in expectation over all possible input permutations.
Randomized Algorithms:

- Analysis is true for any input.
- The sample space is the space of random choices made by the algorithm.
- Repeated runs are independent.

Probabilistic Analysis:

- The sample space is the space of all possible inputs.
- If the algorithm is deterministic repeated runs give the same output.
A **Monte Carlo Algorithm** is a randomized algorithm that may produce an incorrect solution. For decision problems: A **one-side error** Monte Carlo algorithm errs only one possible output, otherwise it is a **two-side error** algorithm.

A **Las Vegas** algorithm is a randomized algorithm that **always** produces the correct output.

In both types of algorithms the run-time is a random variable.