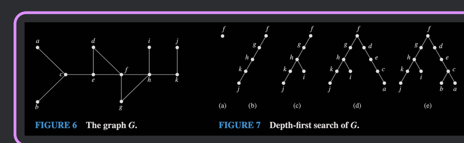
Data Structure

- **Linear Data Structures**
  - **Array**
    Arrays provide O(1) random access but have a fixed size.
  - **Linked List**
    A linked list uses nodes connected by pointers.
    - Singly Linked List
    - Doubly Linked List
    - Circular Linked List
  - **Stack**
    A stack follows the LIFO principle.
  - **Queue**
    A queue follows the FIFO principle.
    - **Deque (Double-Ended Queue)**
      A deque allows insertion and deletion from both ends.
    - **Priority Queue**
      A priority queue processes elements based on priority rather than insertion order.
- **Hash-based Data Structures**
  - **Hash table**
    A hash table provides average O(1) lookup time.

    Key --(Hash function)--> Bucket
    - 处理冲突的方法
      - Separate Chaining
      - Open Addressing
- **Non-linear Data Structure**
  - Graph
    - **Structure**
      Path, Cycle
      Weighted Graph,
      Directed / Undirected Graph
    - Shortest-Path Problems
      - BFS
        - Undirected Graph
      - Dijkstra's Algorithm
        - 带权路径长度
      - Floyd's Algorithm
    - Topological sort
      找到做事的先后顺序
    - **Spanning Trees**
      Let G be a simple graph. A spanning tree of G is a subgraph of G that is a tree containing every vertex of G.
      - **Depth-First Search (DFS)**
        - 
          FIGURE 6 The graph G.
          FIGURE 7 Depth-first search of G.
      - **Breadth-First Search (BFS)**
    - **Minimum spanning tree**
      A minimum spanning tree in a connected weighted graph is a spanning tree that has the smallest possible sum of weights of its edges.
      - Prim's algorithm
        - Begin by choosing any edge with smallest weight, putting it into the spanning tree. Successively add to the tree edges of minimum weight that are incident to a vertex already in the tree, never forming a simple circuit with those edges already in the tree. Stop when n − 1 edges have been added.
      - Kruskal's algorithm
        - choose an edge in the graph with minimum weight. Successively add edges with minimum weight that do not form a simple circuit with those edges already chosen. Stop after n − 1 edges have been selected.
  - Trees
    - Structure
      - Basic concepts
        - Node, Root, Edge
        - Parent, Child, Sibling, Ancestor, Descendant
        - Leaf Node / External Node, Internal Node
        - Subtree, Forest
      - Tree & Node Properties
        - **Degree**
          Degree of a node / tree
        - **Depth**
          Depth of a node
        - **Height**
          Height of a node / tree
        - **Level**
          Level of a node
    - Common Types of Trees
      - Binary Tree
        - Full Binary Tree
          - A binary tree where every node has either 0 or 2 children.
        - Complete Binary Tree
          - A binary tree in which all levels are completely filled, except possibly the last level, and the last level has all nodes as far left as possible.
        - Perfect Binary Tree
          - A full binary tree in which all leaf nodes are at the same level.
      - **Trie Tree**
        A tree-like data structure used to store a dynamic set of strings, where keys are usually strings. It is very efficient for prefix-based searches.
      - Tree for Searching
        - **Binary Search Tree (BST)**
          Left Child value < Root Value < Right Child Value
          - **AVL Tree / Balanced Binary Tree**
            - A binary tree where the heights of the left and right subtrees of any node differ by at most one.
            - 每次调整最小不平衡子树
          - **Red-Black Tree**
          - Splay tree
        - **B Tree**
          m-ary Search Tree
          - [m/2] <= #ary at each node <= m
          - [m/2]-1 <= #key at each node <= m-1
          - Leaf Node: 失败节点
        - **B+ Tree**
          类似分类查找，每个Node key保存subtree最大的key
          - 非叶根Node最少有2棵subtree
          - 其他每个Node至少[m/2]棵subtree
          - Leaf Noise: record
      - Tree for Sorting
        - **Heap**
          A specialized complete binary tree that satisfies the heap property.
          - **Min-Heap**
            The key of a parent node is less than or equal to the keys of its children.
          - **Max-Heap**
            The key of a parent node is greater than or equal to the keys of its children.
    - Tree Traversal (遍历)
      - **Depth-First Search (DFS)**
        - **Pre-order traversal**
          Root -> Left Subtree -> Right Subtree
          - Prefix (or Polish) expression
        - **In-order traversal**
          Left Subtree -> Root -> Right Subtree
          - Infix expression
        - **Post-order traversal**
          Left Subtree -> Right Subtree -> Root
          - Postfix (or reverse Polish) expression
          - Useful for deleting nodes and their subtrees.
      - **Breadth-First Search (BFS) / Level-order Traversal**
        - Visits nodes level by level, from top to bottom, and from left to right within each level.
- **Algorithm**
  - Algorithm Quality
    - Time complexity
      - Best / Worst / Average Case
    - Space complexity
    - Stability
      - 如果待排序的序列中有两个或多个相等的元素，在排序后，这些相等元素的相对位置保持不变，则称该排序算法是稳定的。
  - Sorting Algorithms
    - 插入排序
      - Insertion Sort
        - 和前面的比，找到对应位置插入
      - Shell Sort
        - 分成几个子表，对每个子表进行直接插入排序
    - 选择排序
      - Selection Sort
        - 扫一遍，找到最小的，和头部交换位置
      - Heap Sort
        - 非终端节点i <= [n/2]，把所有非终端节点都检查一遍，若不满足则调整
    - 交换排序
      - Bubble Sort
        - 从后两两对比，更小的放前
      - Quick Sort
        - 选择第一个做pivot，小放pivot左，大放pivot右，直到low和high指针相遇
    - Merge Sort
      - **Divide and Conquer**
        Divide: 将数组递归地对半拆分，直到每个子数组只有一个元素。
        Merge: 将相邻的子数组两两合并，并在此过程中排序。
        需要额外的空间来存储合并过程中的临时数组。
    - Counting Sort

Discrete Probability

- **Random Events**
  - Events
    - p(X U Y) = p(X) + p(Y) - p(XY)
  - Independent Events
    - Definition: p(XY) = p(X)p(Y)
  - Random Variables
    - Independent random variables

      The random variables X and Y on a sample space S are *independent* if

      $$p(X = r_1 \text{ and } Y = r_2) = p(X = r_1) \cdot p(Y = r_2),$$

      or in words, if the probability that $X = r_1$ and $Y = r_2$ equals the product of the probabilities that $X = r_1$ and $Y = r_2$, for all real numbers r_1 and r_2.
  - Uniform distribution
    - Suppose that S is a set with n elements. The uniform distribution assigns the probability 1/n to each element of S.
  - Conditional Probability P(X | Y) = P(XY) / P(Y)
    - **Bayes' Formula**
      P(A|B) = P(B|A)P(A) / P(B)

      Updating beliefs according to new evidence
      - Posterior (后验) P(A|B)
        - The probability of our hypothesis A, given the evidence B.
      - Likelihood (似然) P(B|A)
        - The probability of seeing the evidence B, if our hypothesis A is true.
      - Prior (先验概率) P(A)
        - Our initial belief in the hypothesis A, before seeing any evidence.
      - Evidence (证据) P(B)
        - The overall probability of observing the evidence B.
    - Bayes' Theorem

      BAYES' THEOREM Suppose that E and F are events from a sample space S such that $p(E) \neq 0$ and $p(F) \neq 0$. Then

      $$p(F \mid E) = \frac{p(E \mid F)p(F)}{p(E \mid F)p(F) + p(E \mid \overline{F})p(\overline{F})}.$$

- **Expected Value and Variance**
  - Expected Values E(X)

    The *expected value*, also called the *expectation* or *mean*, of the random variable X on the sample space S is equal to

    $$E(X) = \sum_{s \in S} p(s)X(s).$$

    The *deviation* of X at $s \in S$ is $X(s) - E(X)$, the difference between the value of X and the mean of X.
    - E(aX+b) = aE(X) + b
    - E(X1 + X2 + ... + Xn) = E(X1) + E(X2) + ... + E(Xn)
    - 若X与Y独立，E(XY) = E(X)E(Y)
  - Variance V(X) = E((X-E(X))^2)
    - V(X) = E(X^2) - E^2(X)
    - V(cX) = c^2 V(X)
    - V(X+Y) = V(X) + V(Y) + 2(E(XY) - E(X)E(Y))
      - 若X与Y独立，V(X+Y) = V(X) + V(Y)
  - Covariance (协方差)
    Cov(X,Y) = E((X-E(X))(Y-E(Y)))
    - Cov(X, Y) = Cov(Y, X)
    - Cov(aX, bY) = ab Cov(X, Y)
    - Cov(X+Y, Z) = Cov(X, Z) + Cov(Y, Z)
    - 若X与Y独立，Cov(X, Y) = 0

- **Probability Distribution**
  - (0, 1) distribution X ~ (0, 1)
    - E(X) = p
    - V(X) = p(1-p)
  - Bernoulli distribution X ~ B(n, p)
    - p(X = k) = C(n,k)p^k (1-p)^(n-k)
    - E(X) = E(X1) + E(X2) + ... + E(Xn) = np
    - V(X) = np(1-p)
  - Poisson theorem (λ = np)
  - Poisson distribution X~ P(λ)
    - P(X=k) = λ^k e^{-λ} / k!
    - E(X) = λ
    - V(X) = λ
  - Geometric distribution X ~ G(p)
    - Example: Suppose that the probability that a coin comes up tails is p. This coin is flipped repeatedly until it comes up tails. What is the expected number of flips until this coin comes up tails? 直到首次... 不达目的不罢休
    - Definition: A random variable X has a geometric distribution with parameter p if p(X = k) = (1 − p)^{k−1}p for k = 1, 2, 3, ..., where p is a real number with 0 ≤ p ≤ 1.
  - Hypergeometric distribution X～ H(N,M,n)
    - P(X = k) = C(M, k) C(N-M, n-k) / C(n, N)

- **Limit Theorems**
  - Markov's Inequality
    - If X is a random variable that takes only nonnegative values, then for any value a > 0

      $$P\{X \geq a\} \leq \frac{E[X]}{a}$$
  - Proof 2
  - Chebyshev's Inequality
    - Let X be a random variable on a sample space S with probability function p. If r is a positive real number, then

      $$p(|X(s) - E(X)| \geq r) \leq V(X)/r^2$$
      - Proof 1

        Proof: Let A be the event

        $$A = \{s \in S \mid |X(s) - E(X)| \geq r\}.$$

        What we want to prove is that $p(A) \leq V(X)/r^2$. Note that

        $$V(X) = \sum_{s \in S}(X(s) - E(X))^2 p(s) = \sum_{s \in A}(X(s) - E(X))^2 p(s) + \sum_{s \notin A}(X(s) - E(X))^2 p(s).$$

        The second sum in this expression is nonnegative, because each of its summands is nonnegative. Also, because for each element s in A, $(X(s) - E(X))^2 \geq r^2$, the first sum in this expression is at least $\sum_{s \in A} r^2 p(s)$. Hence, $V(X) \geq \sum_{s \in A} r^2 p(s) = r^2 p(A)$. It follows that $V(X)/r^2 \geq p(A)$, so $p(A) \leq V(X)/r^2$, completing the proof. ◁
    - Chebyshev's inequality provides a universal guarantee for any probability distribution, stating that a certain fraction of its values must be within a certain distance of the mean. Specifically, it guarantees that no more than 1/k² of the data can be more than k standard deviations away from the mean.
  - **Law of Large Numbers (LLN)**
    as the number of trials of an experiment increases, the sample average of the outcomes will converge to the theoretical expected value.
    - Chebyshev's Law (切比雪夫大数定律)
      - X_1, X_2, ..., X_n 相互独立，E(X_i), V(X_i) 存在，V(X_i)\le C，则 \bar{X} \to E(\bar{X})
    - Bernoulli's Law (伯努利大数定律)
      - n_A 为n次伯努利试验中A发生的次数，P(A)=p，则 n_A/n \to p
    - Khinchin's Law (辛钦大数定律)
      - X_1, X_2, ..., X_n 相互独立且服从同一分布，E(X_i) 存在，则 \bar{X} \to E(\bar{X})
  - **Central Limit Theorem** (中心极限定理)
    - 在一定条件下，即使原来并不服从正态分布的一些独立随机变量的和的分布，当n充分大时，也近似服从正态分布

      Theorem 2.2 (Central Limit Theorem) Let $X_1, X_2, \ldots$ be a sequence of independent, identically distributed random variables, each with mean μ and variance σ^2. Then the distribution of

      $$\frac{X_1 + X_2 + \cdots + X_n - n\mu}{\sigma\sqrt{n}}$$

      tends to the standard normal as $n \to \infty$. That is,

      $$P\left\{\frac{X_1 + X_2 + \cdots + X_n - n\mu}{\sigma\sqrt{n}} \leq a\right\} \to \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{a} e^{-x^2/2}\,dx$$

      as $n \to \infty$.

Mathematics

- **Calculus**
  - Basic
    - Domain
      - Rational / Irrational
      - Real / Imaginary
    - Range
  - Limits and Continuity
    - Limit (极限)
    - Continuous
    - Squeeze Theorem (夹逼定理)
  - Derivatives
    - f'(a) = lim_{h->0} (f(a+h) - f(a)) / h
    - Differentiable (可导 / 可微)
      - f'_{-}(a) = f'_{+}(a)
  - Series
    - Taylor Formula
      - $f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(a)}{k!}(x-a)^k + R_n(x)$
      - Core idea: local approximation
- **Linear Algebra**
  - Eigenvalues and Eigenvectors
    - Ax = λx
    - **Geometric Intuition** Eigenvectors are the directions that remain unchanged (up to scaling) by the linear transformation A. Eigenvalues (λ) are the scaling factors in those directions.
  - Matrix Operations
    - Transpose
    - Determinant
      - det(A)=∏λi
    - Inverse / Invertible
    - Rank
      - The rank of a matrix is the maximum number of linearly independent rows (or columns) it contains, which represents the dimension of the vector space spanned by the matrix.
    - Trace
      - tr(A)=∑λi
      - In linear algebra, the most common and significant type of matrix transformation is a similarity transformation. This transformation takes a matrix A and transforms it into $P^{-1}AP$, where P is any invertible matrix.
  - Singular Value Decomposition (SVD)
    - A = UΣV^T
    - Any linear transformation can be broken down into a sequence of three operations: a rotation (V^T), a scaling along the coordinate axes (Σ), and another rotation (U).