- ## Ways of reducing collisions:-

    - □ Spread out the records by choosing a good hash function

    - □ Use extra memory, i.e. increase the size of the address space

    - □ Put more than one record at a single address use of buckets

what's the best (BFS-DFS)?

Breadth-First Search (BFS)

- *Exploration: Explores all nodes at the present depth level before moving on to nodes at the next depth level.*

- *Use Cases: Ideal for finding the shortest path in unweighted graphs, such as in social networks or web crawlers.*

- *Memory Usage: Can consume a lot of memory if the tree/graph is wide.*

- *Time Complexity: Generally (O(V + E)), where (V) is the number of vertices and (E) is the number of edges.*

Depth-First Search (DFS)

- *Exploration: Explores as far as possible along each branch before backtracking.*

- *Use Cases: Suitable for problems like topological sorting, solving puzzles (like mazes), and detecting cycles in graphs.*

- *Memory Usage: Uses less memory compared to BFS if the tree/graph is deep.*

- *Time Complexity: Also (O(V + E)).*

When to Use Which?

- *BFS: Use when you need the shortest path or when the solution is likely to be found at the shallowest depth.*

- *DFS: Use when you need to explore all possible paths or when the solution is likely to be found deep in the tree/graph.*

what is using their binary-tree in software ?

1.Binary Search Trees (BST)

- Applications: Used in many search applications where data is constantly entering and leaving, such as in databases and file systems.

- Example: Implementing map and set objects in many programming languages' libraries.

2. Binary Space Partitioning (BSP)

- Applications: Used in 3D video games to determine which objects need to be rendered.

- Example: Efficiently managing and rendering complex scenes in game development.

3. Heaps

- Applications: Used to implement priority queues, which are essential for scheduling processes in operating systems.

- Example: The heap data structure is used in algorithms like Dijkstra's shortest path and in heap sort.

4. Huffman Coding Trees

- Applications: Used in data compression algorithms.

- Example: Compression formats like JPEG and MP3 use Huffman coding to reduce file sizes.

5. Syntax Trees

- Applications: Used by compilers to parse expressions and generate executable code.

- Example: Abstract Syntax Trees (AST) are used in the compilation process of programming languages.

6. Trie (Prefix Tree)

- Applications: Used for efficient retrieval of keys in a dataset of strings.

- Example: Implementing autocomplete features in search engines and text editors.

7. Game AI

- Applications: Used to implement decision-making processes in game AI.

- Example: Representing possible moves in a game and searching for the best move.

8. File Systems

- Applications: Used to manage hierarchical data structures.

- Example: Directory structures in operating systems.