UNIT - 4

DECISION TABLES

Meaning: Decision tables are a structured way to represent complex decision-making processes. They help in organizing and visualizing different conditions and their corresponding actions or outcomes.

Representation: Decision tables are typically represented in a tabular format. They consist of rows and columns where each row represents a unique combination of conditions, and each column represents a specific condition or action.

Implementation: Decision tables can be implemented using various tools like spreadsheets, software applications, or even manually on paper. In implementation, you list out all the possible conditions that could affect a decision, and then determine the corresponding actions or outcomes for each combination of conditions.

Examples: Let's say you're designing a decision table for deciding whether to go for a picnic based on weather conditions. You might have conditions like "Is it raining?", "Is it windy?", and "Is it sunny?" The actions could be "Go for a picnic" and "Stay indoors." Your decision table might look like this:

Weather	Is it raining?	Is it windy?	Is it sunny?	Action
Decision 1	Yes	No	No	Stay indoors
Decision 2	No	Yes	No	Stay indoors
Decision 3	No	No	Yes	Go for a picnic
Decision 4	Yes	Yes	No	Stay indoors
Decision 5	Yes	No	Yes	Stay indoors
Decision 6	No	Yes	Yes	Stay indoors
Decision 7	Yes	Yes	Yes	Stay indoors
Decision 8	No	No	No	Stay indoors

In this table, each row represents a different combination of weather conditions, and the corresponding action to take.

So, decision tables are a handy tool for organizing complex decision-making processes into a clear and structured format, making it easier to understand and implement decisions.

Applications of Decision Tables:

1. **Business Rules Management:** Decision tables are commonly used in business rules management systems to model complex decision-making processes in industries such as finance, insurance, and healthcare. They help organizations manage and automate business rules efficiently.

2. **Software Testing:** Decision tables are used in software testing to represent different test scenarios and expected outcomes. Testers can use decision tables to ensure comprehensive test coverage and identify potential edge cases.

3. **Control Systems:** Decision tables find applications in control systems, where they are used to define rules and conditions for controlling various processes or devices. They help in optimizing and automating control processes in industries like manufacturing and automation.

4. **Medical Diagnosis:** Decision tables are used in healthcare for medical diagnosis and treatment planning. They help healthcare professionals analyze patient symptoms, medical history, and test results to make informed decisions about diagnoses and treatment options.

5. **Quality Assurance in Manufacturing:** Decision tables find applications in quality assurance processes in manufacturing industries. They are used to define quality control rules and criteria for inspecting and testing products, ensuring adherence to quality standards and regulations.

SYMBOL TABLES

Meaning: A symbol table is a data structure used by compilers, interpreters, and other software tools to store information about symbols used in a program. Symbols can include variable names, function names, constants, and data types.

Representation: Symbol tables are often implemented as key-value pairs, where the key is the symbol (e.g., variable name) and the value is information associated with that symbol (e.g., data type, memory location). They can be represented using hash tables, binary search trees, or other data structures.

Implementation: Symbol tables are implemented as part of a compiler or interpreter. When a program is compiled or interpreted, the symbol table is built and updated as the program is parsed. Information about each symbol, such as its name, data type, scope, and memory location, is stored in the symbol table for reference during compilation or execution.

Examples: Let's consider a simple example in a programming language like Python:

```
# Symbol Table Example in Python
# Function definition
def calculate_area(radius):
    pi = 3.14  # Variable declaration and initialization
    area = pi * radius * radius  # Calculation
    return area

# Main program
radius_value = 5
result = calculate_area(radius_value)  # Function call
print("Area of the circle:", result)
```

In this example:

- The symbol table would store information about symbols such as **`calculate_area`**, **`pi`**, **`radius`**, **`area`**, **`radius_value`**, **`result`**, etc.
- Each symbol would have associated information like its name, data type, scope, and memory location.
- For example, **`radius_value`** would have information about its data type (integer), its value (5), and its memory location.

Symbol tables play a crucial role in the compilation and execution of programs by providing a centralized repository of information about symbols used in the code. They help ensure that variables are declared and used correctly and facilitate efficient memory management and optimization during compilation or interpretation.

Applications of Symbol Tables:

1. **Compilers and Interpreters:** Symbol tables are crucial components of compilers and interpreters. They are used to store information about variables, functions, and other symbols encountered during the compilation or interpretation process.

2. **Static Code Analysis:** Symbol tables are used in static code analysis tools to perform various analyses on source code, such as detecting unused variables, identifying variable scope issues, and enforcing coding standards.

3. **Debugging:** Symbol tables are helpful in debugging programs by providing information about the symbols used in the code, such as variable names and their values, function names, and data types. Debuggers use symbol tables to inspect the state of a program during runtime.

4. **Dynamic Linking and Loading:** Symbol tables are essential for dynamic linking and loading of libraries and modules in software systems. They store information about functions and variables used in external libraries, facilitating their integration into a program at runtime.

5. **Symbolic Mathematics:** Symbol tables are used in symbolic mathematics software to manipulate algebraic expressions, equations, and mathematical symbols. They store information about mathematical variables, constants, and operations, enabling symbolic computation and mathematical analysis.

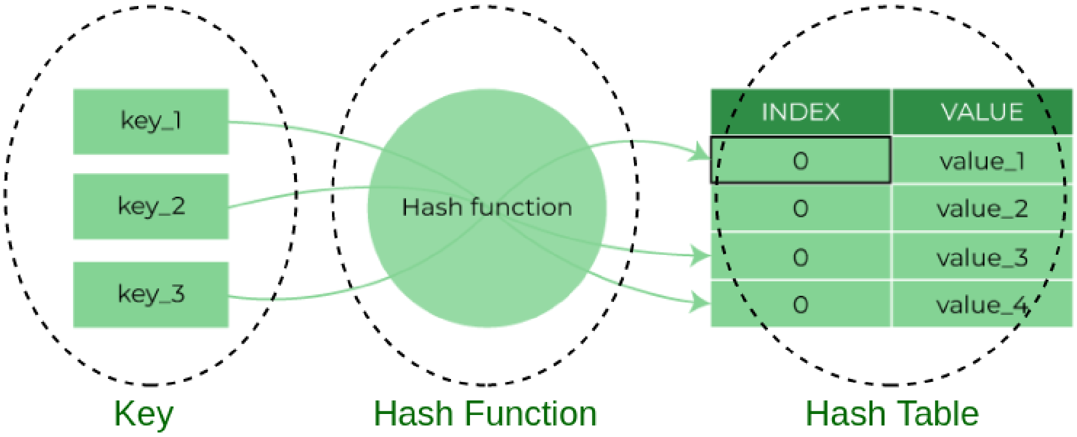
HASH TABLES

Meaning: A hash table is a data structure used to store and retrieve key-value pairs efficiently. It offers fast insertion, deletion, and lookup operations, making it ideal for tasks like indexing, caching, and storing data in memory.

Representation: Hash tables typically consist of an array of buckets or slots, where each slot can store a key-value pair. To determine the position of a key-value pair within the array, a hash function is used. The hash function takes the key as input and computes a hash code, which is then mapped to a specific slot in the array.

Implementation: In implementation, a hash table consists of three main components: the array, the hash function, and a collision resolution mechanism.

- The array provides the storage for key-value pairs.
- The hash function calculates the hash code for each key, determining its position in the array.
- Collision resolution mechanisms handle situations where two keys produce the same hash code, ensuring that each key-value pair is stored correctly.



Hash tables are widely used in computer science and programming due to their efficiency in storing and retrieving data. They are employed in various applications such as databases, caching mechanisms, and hash-based data structures like sets and dictionaries.

Applications of Hash Tables:

1. **Data Storage and Retrieval:** Hash tables are widely used in data storage and retrieval applications, such as databases, caching systems, and associative arrays. They offer fast insertion, deletion, and lookup operations, making them suitable for storing and accessing large volumes of data efficiently.

2. **Hash-Based Data Structures:** Hash tables serve as the underlying data structure for implementing other hash-based data structures like sets, dictionaries, and hash maps. These data structures are used in various algorithms and applications for storing and organizing data.

3. **Cryptographic Applications:** Hash functions derived from hash tables are utilized in cryptographic applications for generating hash values, digital signatures, and message authentication codes. Hash functions play a critical role in ensuring data integrity and security in cryptographic protocols.

4. **Network Routing:** Hash tables are used in network routing algorithms to efficiently store and retrieve routing information, such as IP addresses and routing paths. They enable fast lookup of routing tables, contributing to efficient packet forwarding and routing in computer networks.

5. **Spell Checking and Auto-Complete:** Hash tables are employed in spell-checking and auto-complete features in text editors, word processors, and search engines. They store dictionaries and vocabulary lists, enabling fast lookup of words and suggestions based on partial inputs or misspelled words.