The decision trees are not always the most appropriate and the best tool for the decision making process. Representing a very complex system with this tool may lead to a huge number of branches with a similar number of possible paths and options.For a complex problem, analysing various situations is very difficult and can confuse the analyst.

Decision tree is a method for defining complex relationships by describing decisions and avoiding the problems in communication. Through pictorial presentation, it shows alternative actions and conditions within horizontal tree framework. Thus, it depicts which conditions to consider first, second, and so on.

Further, decision tree depicts the relationship of each condition and their permissible actions as shown in figure 6.7. A square node indicates an action and a circle indicates a condition. It forces analysts to consider the sequence of decisions and identifies the actual decision that must be made.

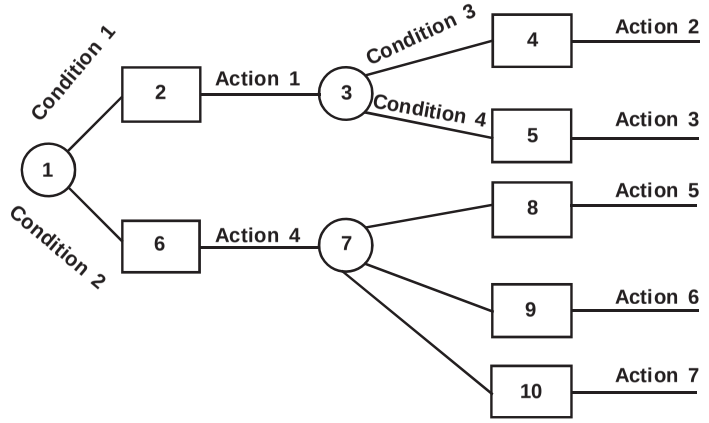


Figure 6.7:The relationship of condition and its corresponding action in decision tree

The major limitation of a decision tree is that it lacks information in its format to describe what other combinations of conditions you can take for testing. It is a single representation of the relationships between conditions and actions. Another example of a decision tree is presented in figure 6.8 describing discount policy of a super market store.

Pros and cons of data dictionaries:

Data dictionaries can be a valuable tool for the organization and management of large data listings. Fundamentally, data dictionary includes the following pros:

- Provides organized, comprehensive list of data
- Easily searchable
- Can provide reporting and documentation for data across multiple programs
- Simplifies the structure for system data requirements
- No data redundancy
- Maintains data integrity across multiple databases
- Provides relationship information between different database tables
- Useful in the software design process and test cases

Though they provide thorough listings of data attributes, data dictionaries may be difficult to use for some users. Other cons include:

- Functional details not provided
- Not visually appealing
- Difficult to understand for non-technical users



CHECK YOUR PROGRESS

Q.2: Select the correct option for the following questions:

i) The data dictionary in SDLC contains descriptions of
   (a) DFD elements (b) ER Diagram
   (c) Class Diagram (d) Decision table

ii) A data dictionary is useful as
   (a) it is a documentation aid
   (b) it assists in designing input forms
   (c) it contains all data in an application including temporary data used in processes
   (d) it is a good idea in system design

6.5 STRUCTURED ENGLISH

Structured English is derived from structured programming language that gives a more understandable and precise description of the process. Structured English is a tool for structured analysis that refers to the use of the English language with the syntax of structured programming to communicate to non-technical users by breaking down the program into logical steps using straight-forward English words. This tool aims to get the benefits of both programming logic and natural language. Programming logic helps to achieve precision, while natural language helps with the familiarity of the spoken word.

Most programmers are unaware of the large picture of software due to which they only rely on what their managers tell them to do. It is the responsibility of higher software management to provide accurate information to the programmers to develop accurate yet fast code. Different methods which present a programming logic graphically might be interpreted differently by different people. Structured English uses plain English words in the structured programming paradigm. It is not the ultimate code but a kind of description of what is required to code and how to code it. The following are some tokens of structured programming:

IF-THEN-ELSE, DO-WHILE-UNTIL, etc.

These tokens are used by analysts and designers of the software with Structured English. It is nothing but the description of what is required to code and how to code it. Structured English helps the programmer to write error-free code with the following.

- It is best used when sequences and loops in a program must be considered and the problem needs sequences of actions with decisions.
- It does not have strict syntax rule. It expresses all logic in terms of sequential decision structures and iterations.

For example, the following sequence of actions are written using Structured English

In our day-to-day life, many a times we have to face various complex cases where the most appropriate action under several conditions cannot be determined easily and for such a case a decision tree is a great aid. Hence, this decision tree representation is very effective in describing the business problems involving more than one dimension and parameters.They also point out the required data, which surrounds the decision process. All the data used in the decision making should be first described and defined by the analyst so that the system can be designed to produce correct output data.

Let us consider an example. The discount policy of a saree manufacturer for his customers is fixed as described in the following. According to the policy, the saree manufacturer offers discount to his customers based on the type of customer and size of their order. For the individual, only if the order size is 12 or more, the manufacturer gives a discount of 50% and for less than 12 sarees the discount is 30%. Whereas in case of shopkeeper or retailers, the discount policy is different. If the order is less than 12 then there is 15% discount. For 13 to 48 sarees order, the discount is 30%, for 49 to 84 sarees 40% and for more than 85 sarees the discount is 50%. The decision policy for discount percentage can be put in the form of a decision tree which is presented in the following figure 6.6.

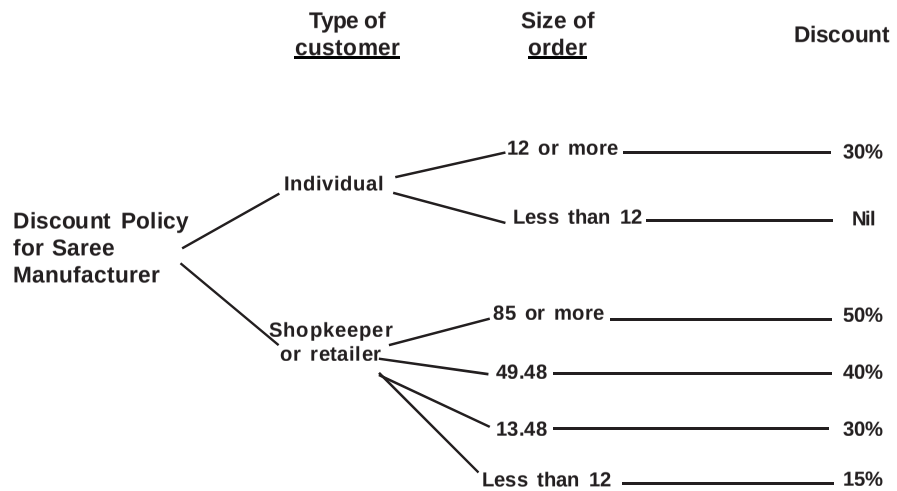


Figure 6.6: Decision tree presenting the discount policy of Saree Manufacturer

allows the analyst to write on the branches to describe conditions and actions of a logic. In constructing a decision tree, a square is used to indicate an action and a circle is used to represent a condition.

The decision sequence starts from the root of the tree that is usually on the left of the diagram. The path to be followed to traverse the branches is decided by the priority of the conditions and the respectable actions. A series of decisions are taken, as the branches are traversed from left to right. The nodes are the decision junctions. After each decision point there are next set of decisions to be considered. Therefore, at every node of the tree represented conditions are considered to determine which condition prevails before moving further on the path.

This decision tree structure is very beneficial to the analyst. The first advantage of decision tree is that the analyst will be able to depict all the given parameters in a logical format which enables the simplification of the whole decision process with a very remote chance of committing an error in the decision process as all the options are clearly specified in one of the simplest manner.

Secondly, it also helps the analyst about those decisions, which can only be taken when couple or more conditions should hold true together for there may be a case where other conditions are relevant only if one basic condition holds true. A schematic structure of the decision tree is shown in the figure 6.5.

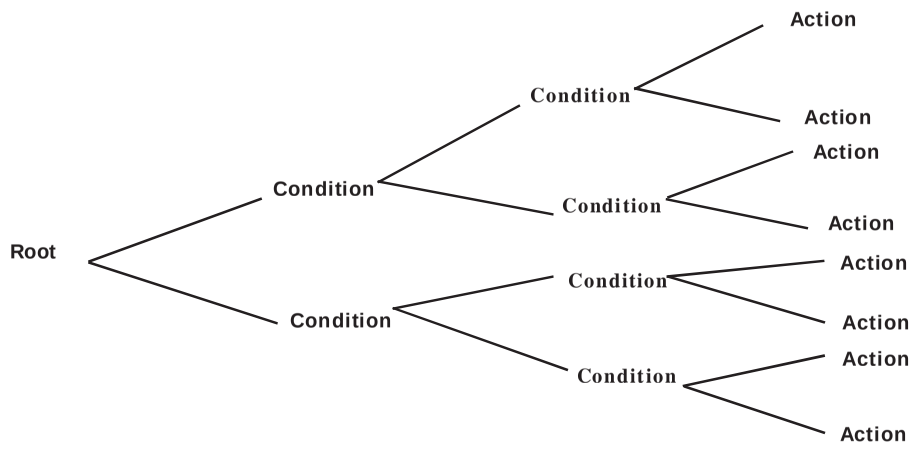


Figure 6.5:Schematic structure of the decision tree


```
if customer pays advance
  then
  Give 5% Discount
else
  if purchase amount >=10,000
    then
      if the customer is a regular customer
        then Give 5% Discount
      else No Discount
    end if
  else No Discount
end ifend if
```

Writing Structured English:

In order to write structured English, the following conventions can be used:

1. Express all logic in terms of one of four types-sequential structure, decision structure, case structure, and iteration.
2. Use capitalized keywords like IF, THEN, ELSE, DO, DO WHILE, DO UNTIL & PERFORM.
3. Indent blocks of statements to present their hierarchy (nesting) clearly.
4. When words or phrases are defined in a data dictionary, underline those words or phrases to signify that they have a specialized meaning.
5. Be careful when using “and” and “or” and avoid confusion when distinguishing between “greater than” and “greater than or equal to” and like relationships “A and B” means A and B whereas “A or B” indicates either A or B.

Example

Let us consider another example of Customer Authentication in the online shopping environment. This procedure to authenticate customer can be written in Structured English as:

Action

```
Enter Customer_Name
SEEK Customer_Name in Customer_Name_DB file
IF Customer_Name found THEN
Call procedure USER_PASSWORD_AUTHENTICATE()
ELSE
PRINT error message
Call procedure NEW_CUSTOMER_REQUEST()
ENDIF
```

The code written in Structured English is more like day-to-day spoken English. It cannot be implemented directly as a code of software. Structured English is independent of programming language.

The objectives of using Structured English are as follows:

a) to describe computational procedures reasonably precisely which can be understood by any user;
b) to expand a DFD so that a user can understand it;
c) to develop algorithms corresponding to processes in a DFD;
d) to ease writing programs for DFDs.



CHECK YOUR PROGRESS

Q.3: Find out the correct option for the following Multiple choice questions:

i) Structured English statements must be
   a. short and clear
   b. specified quantitatively
   c. specified qualitatively
   d. detailed and descriptive

ii) Select correct decision structures from the following:

```
A. if total marks>=75
        then
        enter distinction in student record
  end if
```



```
B. if total marks >=50
        then
        enter pass in student record
  else
        enter fail in student record
  end if
C. if total marks >=60
then
        enter first class in student record
  else if total marks>= 50
        then
        enter second class in student record
  else
        enter fail in student record
  end if
  end if
D. if attendance <30%
        then
        Do not admit in examination
  else
```

a. B and C b. A and B
c. C and D d. A and D

6.6 DECISION TREE

Decision tree is a tree like structure that represents the various conditions and the subsequent possible actions. It also shows the priority in which the conditions are to be tested or addressed. Each of its branches stands for any one of the logical alternatives and because of the branch structure, it is known as a tree. Decision trees are employed when complex branching occurs in a structured decision process. Although it derives its name from natural trees, but it is drawn keeping root of the tree on the left side and branches coming out from the root to the right side. In this way, it