# Experiment No. 8

**Case Study on Expert Systems**

**Name:Rebecca Dias Roll no./PID: 19/182027**

**Aim:** To study and understand the working of expert systems.

# Theory:

Expert Systems are computer programs built for commercial application using the programming techniques of artificial intelligence.

**Examples:** There are many examples of expert system. Some of them are given below:

* MYCIN: One of the earliest expert systems based on backward chaining. It

can identify various bacteria that can cause severe infections and can also recommend drugs based on the person’s weight.

* DENDRAL: It was an artificial intelligence based expert system used for chemical analysis. It used a substance’s spectrographic data to predict it’s molecular structure.
* R1/XCON: It could select specific software to generate a computer system wished by the user.

# Features of Expert systems

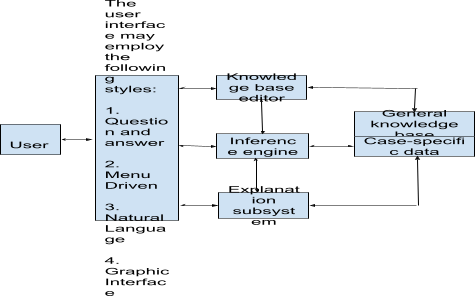
* + Facility of solving problems that require an expertise
  + Speedy Solutions
  + Reliable Solutions
  + Cost Reduction
  + Elimination of uncomfortable and monotonous operations
  + Power to manage without human experts
  + Wider access to knowledge

# When to use Expert Systems?

* + Human experts are difficult to find
  + Human experts are expensive
  + Knowledge improvement is needed
  + Knowledge is difficult to acquire
  + Poor available information
  + Problem is subjected to change

# Basic Elements of Expert System

An expert system is typically composed of at least three primary components. These are the inference engine, the knowledge base, and the user interface.



1. **Knowledge Base :** The knowledge base is a collection of rules or other information structures derived from the human expert. Rules are typically structured as If/Then statements of the form:

IF <antecedent> THEN <consequent>

The antecedent is the condition that must be satisfied. When the antecedent is satisfied, the rule is triggered and is said to "fire". The consequent is the action that is performed when the rule fires.

1. **Agenda :** When rules are satisfied by the program, they are added to a queue called the agenda. The agenda is an unordered list of all the rules whose antecedents are currently satisfied. Knowledge bases are typically not ordered, because order tends to play very little role in an expert system. Rules may be placed on the agenda in any order, and they may be fired in any order once they are on the agenda.
2. **Inference Engine :** The inference engine is the main processing element of the expert system. The inference engine chooses rules from the agenda to fire. If there are no rules on the agenda, the inference engine must obtain information from the user in order to add more rules to the agenda. It makes use of knowledge base, in order to draw conclusions for situations. It is responsible for gathering the information from the user, by asking various questions and applying it wherever necessary. It seeks information and relationships from the knowledge base and to provide answers, predictions and suggestions the way a human expert would.
3. **User Interface :** A user interface is the method by which the expert system interacts with a user. These can be through dialog boxes, command prompts, forms, or other input methods.

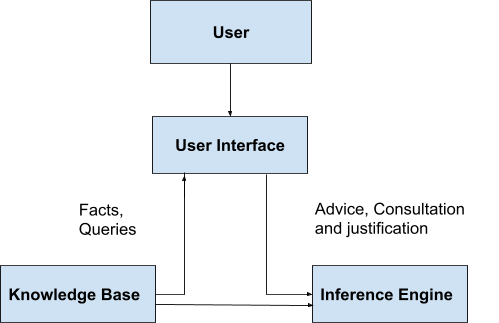
# Working Memory (Database/ Case Specific Data)

Working memory contains the data that is received from the user during the expert system session. Values in working memory are used to evaluate antecedents in the knowledge base. Consequences from rules in the knowledge base may create new values in working memory, update old values, or remove existing values.

# Explanation Mechanism

The method by which an expert system reaches a conclusion may not be obvious to a human user, so many expert systems will include a method for explaining the reasoning process that leads to the final answer of the system.

# Architecture of an Expert System

The architecture of expert system reflects the knowledge engineers way of knowledge representation and performance of intelligent decision making.

This architecture is independent of computer hardware. The user interface allows the system users to

* 1. Enter rules and facts about a particular situation
  2. Ask questions of the system
  3. Provide responses to the user requests
  4. Support all other communication between the system and the user

The knowledge is contained in codified form within the KB which can be easily read and understood. The inference engine uses the information provided to it by the KB and the user to infer new facts.

This architecture can be expanded by extending KB into a knowledge database and domain database. The knowledge base contains rules and domain database contain facts. The KB is updated so that the system provides the most relevant and complete assistance to the user. Some expert system dedicate a module to update the KB known as Knowledge Acquisition Facility. It provides a dialogue with human experts to acquire knowledge in terms of facts and rules and place them in the domain database and the knowledge database respectively.

Self Training Facility: This facility accepts the facts developed by the inference engine and compares derived facts with that in the domain database and if the knowledge is new, it is upgraded.

# Stages in the development of an Expert System

The different stages in the development of an expert system are described as below:

* + 1. **Outline Statement:** This stage identifies an appropriate system. The expert and knowledge engineer work out the concepts, boundaries, relationships and control mechanisms to be included in the

system.Development strategies, constraints and user expectations are explored. The results can be documented in an outline specification for initial prototype development

* + 1. **Knowledge Acquisition :** The experts and knowledge engineers interact intensively in this stage. The experts highlight the essential issues and knowledge engineer tries to comprehend the essence of the knowledge, its limits and its complexities.
    2. **Knowledge Representation:** Once the knowledge engineer has enough knowledge about the system, he has to design a method for appropriate knowledge representation. It is knowledge engineer’s ability to enter into the frame of reference of both expert and user and to suitably structure the acquired knowledge.
    3. **Prototype Development :** Most systems develop a prototype model. The advantage of developing prototype is that we will know whether the system is feasible or not. The users get an opportunity to test the system and whether it is likely to meet their requirements. The developers also gets an opportunity to evaluate the cost and performance of the chosen system.
    4. **Testing:** Testing involves evaluating the performance and utility of the prototype program and revising it as necessary. The prototype should be tested on many problems to evaluate its performance and utility. It may uncover problems such as missing concepts and relations in the representation scheme, knowledge represented at the wrong level of detail.,or infeasible control mechanism. The developers may recycle through different development phases by reformulating the concepts, refining the inference rules and retesting the control flow.
    5. **Main Knowledge Acquisition:** Once the prototype is reviewed and tested, the actual system is developed. This stage assess the extent of the knowledge that is required to meet users needs.
    6. **Specification with Detailed Information:** The detailed specification covers the objectives of the expanded system, the resources required, the

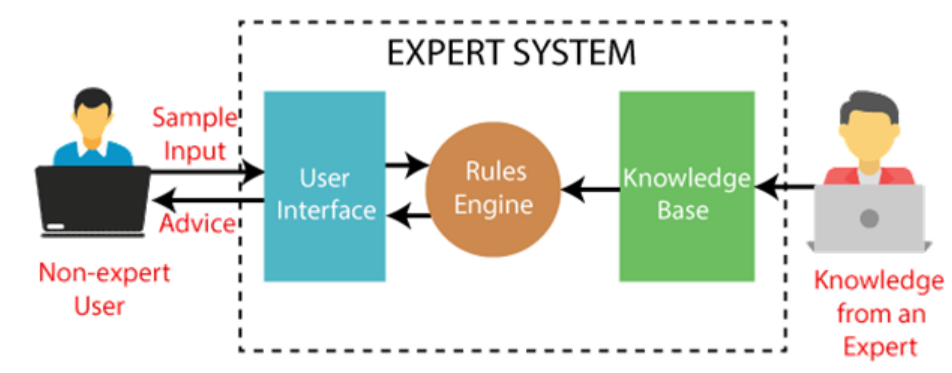
projected time required for implementation, planned costs, system testing and implementation planning.

* + 1. **System Development:** During this stage, it is very important for the users to know exactly how the system is progressing, any problems encountered, and the evidence of new limitations and opportunities. This stage requires careful monitoring.
    2. **Implementation:** Implementation procedures should be carried out by the user and supported by the expert. The implementation plan should have been documented during the specification stage.
    3. **Maintenance:** It requires continuous revision and updating to ensure that the knowledge it contains is always up to date and in accordance with the changing environment in which the organization operates.

**Experiment Exercise :** Identify an expert system and write note on the User Interface, Inference Engine and Knowledge Base of the System

**CaDet EXPERT SYSTEM**

Cancer risk evaluation and early detection are subject to serious limitations mainly related tohuman factors and to characteristics of the data involved. To help overcome these problems, a computer-basedsystem was designed to provide the physician with a clearer clinical picture and aid in directing patients toappropriate measures. Clinical and epidemiological data related to early cancer detection and to cancer risk factors was collected from the literature and incorporated in a database, together with heuristic rules for evalu-ating this data. Individual data obtained from patients through a questionnaire are input into CaDet, a comput-erized clinical decision support system. A report summarizing patient data and cancer hypotheses, with a scoringsystem that reflects degrees of alarm, is generated. The CaDet system, as well as some preliminary results of theclinical experience accumulated in its use, are described. These preliminary results suggest that the approach maybe useful in improving cancer risk assessment and screening in primary care setups.



**USER INTERFACE:**

• Individual data from patients via questionnaire are input into cadet which is computerized decision support system.

• A report summarising patient data and cancer hypothesis with the score is obtained.

• Textbook of medicine W H O cancer statistics data and medline search is used as data source for cadet.

• The cadet procedure has undergone a preliminary trials in US and Israel it seems reasonable and is recommended in use in healthy population above age of 40.

The CaDet computer program is designed to help primary care physicians detect cancer earlier.

• It analyzes the epidemiologic and clinical attributes of individual patients and then identifies in this information and presents to physicians patterns that may require further attention.

• This study was conducted to evaluate the performance of the CaDet program as a tool in early cancer detection.

**INTERFACE DESIGN:**

The CaDet computer program utilizes a multipass heuristic algorithm to pinpoint, weigh, and score po-tential cancers in the patient. It performs a number of iterations on the patient’s responses. Each pass gen-erates a series of different parameters for furtheranalysis by the next pass.The algorithm operates in five phases:

1. Cancer Hypothesis Generation

The patient’s responses arechecked against the database. A list of all cancer hy-potheses possibly related to these responses is gener-ated. This search is continued until all responses arechecked against all cancers, however remote the prob-ability of their presence in the patient.

1. Hypothesis Rejection

All of the patient’s mani-festations are rechecked to ensure the validity of allthe cancers in the possible cancers list. The algorithmattempts to disprove cancers in the list and thus re-move them from further investigation. The systemalso tests the list of possible cancers to ensure thatnone of the cancers listed due to existing maifesta-tions have to be automatically excluded due tophysical impossibility, e.g., testicular cancer in a fe-male, or cancer in an organ which has been previouslyexcised.

1. Initial Scoring Heuristics

 Initial tallies for both HCFand FCF scores are put together. This is done throughtwo sets of heuristics, one for each type of score. Bothscores are simultaneously calculated by analyzingeach manifestation versus the cancer being tested. Toproduce the initial raw weights for FCF and HCFscores, heuristics used at this stage combine arithme-tic and statistical analysis.

1. Secondary Scoring Heuristic

The scores ob-tained in the previous pass are refined by analyzing considerations are considered, as well as the effects of different external affecting factors. The next step of this secondary set of heuristics eliminates cancers thathave too low a likelihood for display. The scores arerefined by a ‘‘filter’’ which produces the final valuesto be displayed in the report. The final list of possiblecancers and scores is then passed on to the final stageof the CaDet algorithm.

1. Supplementary Information

On the basis of the list of possible cancers and theassociated scores, Phase 5 selects recom-mendations and information items from the databasefor display.

**KNOWLEDGE BASE:**

• Factual knowledge:

This knowledge includes epidemiology data risk factor for each cancer site

symptoms and physical sign.

• heuristic knowledge:

Technologies include developed which are characterized quantitatively

clinical Complex cities in several domains including age and sex

dependencies and disease specific interaction among various elements of

data.

• knowledge representation and acquisition:

Each piece of data was assigned a weight reflecting its productor power in

diagnosis of each cancer. Knowledge acquisition refers to process of

extracting structuring and organising domain knowledge from domain

experts into a program for example doctors in case of cadet expert system.

• learning module:

The function of this component is to allow the expert system to acquire

more and more knowledge from various sources and store it in knowledge

base.

**Post Experiment Exercise :**   
Expert systems have been constructed using various general-purpose programming languages as well as specific tools. **LISP and PROLOG** have been used widely. OPS-5 has also been popular among rule-based programmers. OPS is a product of the Instructable Production System Project

**Conclusion:**

In a traditional computer program the logic is embedded in code that can typically only be reviewed by an IT specialist. With an expert system the goal was to specify the rules in a format that was intuitive and easily understood, reviewed, and even edited by domain experts rather than IT experts.

The CaDet computer program seems well adapted to the system's task of aiding physicians to improve early detection. Clinical testing of the system is needed.