الخلاصة

تم في هـذا البحـث تصـميم وتنفيـذ نظـام خبـير لتشـخيص الامـراض الجلديـة العديمـة الصـبغة مـع طـرق المعالجة. تم استخلاص المعرفة المطلوبة من خبيرة المجال (الدكتورة رباب نوري السعدي اخصائية الامراض الجلدية والتناسلية) وكتب طبية في المجال.

يتكـون النظـام مـن خمسـة اجـزاء مسـتخدماً القواعـد الانتاجيـة في تمثيـل المعرفـة. يسـتخدم جـزء ماكنـة الاسـتدلال سـتراتيجية التسلسـل الخلفـي لتمثيـل سـتراتيجية السـيطرة وطريقـة بحـث العمـق ابتـداءً لتمثيـل ستراتيجية البحث. يستطيع النظام ان يعطي استنتاجه مع حالة عدم التأكد وبأستخدام عوامل الموثوقية.

النظام يدعم نوعين من التفسيرات : لماذا وكيف. لغرض فهم كيفية الوصول الى النتائج.

تم انجاز العمل بأستحدام لغة (Visual Prolog) لبناء الواجهة الخبيرة وبناء محرك الستدلال مع ٢٠ قاعدة لبناء قاعدة البيانات.

دار 22 - زقاق 25 - محلة 634 - حي الفردوس-العامرية - بغداد

البريد الالكتروني:-

braq_fadhel@Yahoo.com

لاجل ايجاد معلومات عن موضوع الاطروحة استعمل الكلمات التالية:-

Expert System
Inference Engine
Backward Chaining
Depth First Search
B+_tree
Reasoning under uncertainty

Acknowledgments

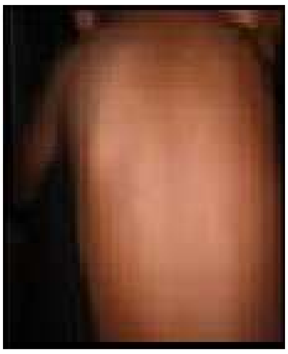
I would like to express my sincere gratitude and appreciation to my supervisors **Dr. Ban Nadeem Thannoon** and **Dr. Rabab Saady** for their valuable guidance, supervision and their efforts during the development of this study.

Grateful thanks for the Department of Computer Science's stuff and employees for their support during study.

Special thanks to my friends and family for their continuous support and encouragement during the period of my study.

Piebaldism Disease

Neavus Depigmentous Disease



ITO Disease

Tuber Sclerosis Disease

Viteligo Disease

Diseases name	Symptoms	Treatments
ALEZZANDRINIS Syndrom	- facial vitiligo - deafness - unilateral tapetoretinal degeneration	
ITO	- irregular shap macule sharply outlined margines - bilateral - CNS abnormalty - abnormal hair - marble cake pattern - abnormal eyes - abnormal teeth - arm and leg length discrepancy - decreased sweat - in(trunk, extremities) sites	no treatment is available diagnosis is for genetic counselling
Oncocercias	- the hypomelanosis gives the patient a leopard liknapearance - the microfiliaria of oncnocerca volvos in skin - the microfliliaria of oncnocerca volvus in blood sputum urine in heavely infected - in(groin, pelviceand axillary area) sites	Ivermectin or surgery for nodules on skin and head
Pinta	- apear as erythematous psoriasi from plaque - lesion be erythematous or copper colour then stated blue - mottled with hypo or hyper pigmentation - symetrically on elbows, kuncklues, knees, ankles, flexor aspect of the wrists	Peniciline
Tinea versicolour	- raised erythematous margines - scales - erythema - has scales when itching - raised - increased in number and size	- 2.5% selenium sulphide - shampoo - oral azole antifungols - cure rates range from 90 to 100% with ketoconazole and itraconazole
VoGT-KoYANA GI-HARADA	- vitiligo - alopecia - poliosis - photophobia - irritation - occular pain - headach - fever - malaise	abundant poliosis remains an indicator of a poor prognosis for repigmentation

	- havsea and vomiting - psychosis - hemiparesis - parafegia - dysphagia - nuchal rigidity	
Ziprkowshi Margolis syndrom	- feather margins - defness - mutism - hetrochromic iridies - in(mid arm, legs, feet, trunk, sparing dorsal spine, abdomen	
histidinemia	- blond hair - blue eyes - CNS abnormality - anemia - elevated blood and urinary histidine level - positive ferric chloride test	
Menkes's steely hair	- buffy face - abnormal hair - progressive cerebal degeneration - bony changes - hypothermia - arterial rupture - arterial thrombosis - defect in copper transport	copper replacement therapy has not proved useful
phenylktonurea	- blond hair - blue eyes - CNS abnormality - eczematous changes - selerodermatous changes - deficiency of Phenylanin hydroxy Lase	low phenylatamine diet results increased pigmentation and often clearing of eczema
Post Kala Azar	- lesions single or confluent to give a map like appearance - cutaneous feature include erythematous macules and yellow to pink nodules - the leishmania-do novani detected in smears or cultures prepared from bone marrow biopsies or spleen aspirates - in(chest, black, arms, anterior thighs, skin from waist to feet is conseicuously spared	pentavalent antimony compounds is effective against vesceral leishmaniasis but hypomelanosis is irreversible
Tuberous sclerosis	- milk white macule - dicrete margins - Ash-leaf spots (long axis of lance ovate macules usually axial on extremities and transverse on trunk	no Rx for the white macules of tubrous sclerosis

	- poliosis - shagreen patch - periungual or subungual fibroma - intraoral fibroma - diffuse bronzing - symmetrically from wrists to elbows or from ankles to knees	
WAARDENBURG's Syndrom	- milk white macule - feather margins - white forelock - displaced medial canthus - premature greying of hair - hypopigmented irides - broad nosal root - eyes brown hyperplasia - defness - in(forehead, neck, anterior chest, abdomen, anterior knees and arm, dorsal hand	- no treatment for the pigmentary diluation is available - spontaneous disappearance of the white forelock - spontaneous repigmentation and contraction of white macules have been reportyed
homocystinurea	- blond hair - blue eyes - CNS abnormality - occular changes - skeletal abnormality - thromboembolic - evaluated blood and urinary homocystine level	- methionine restruction - dietary systeine supplementation - pyridonine suplimentation with systalhioine synthetase deficiency to reversible of hypopigmentation of hair
Neavus depigmentosus	- feather margins - off white macule - no. of melanocytes is normal but dopa reactivity is readuced seizures - limphypertrophy - unilateral - quasidermatomal hypomelanosis - "in(trunk, lower abdomen, face - melanosomes are normal in size or shap , internal structure - melanisation is normal to decrease - melanosomes are occasionally aggregated in melanocytes and in reduced no. in keratinocytes	no available Rx, it is stable
Piebaldism	- milk white macule - feather margines - white forelock - hyperpigmented macules 1cm in diameter within white macules	- sunscreans are adviced - cosmetics and dyes are useful - topical steroids however are not

	- in(mid arm, legs, feet, trunk sparing dorsal spine, abdomen - mucosal involvement - hetrochromic iridies - defness	helpful - PUV both topically and orally has proved disappointing
Tietz syndrom	- blond hair - blue eyes - eye brown hyperplasia - deaf and mutisim	
Vitiligo	- milk white macule - scalop margines - round or oval - hair involvment leukotrichia - halo navus - alopecia areata - aphilhalmological changes - iris abnormal - retinal abnormal - choroidal abnormality - healed chorioretinities	- Topical steroids/ PUVA oral PUVA - 20% MBEH cream to remove normal skin colour
Yaws	- pain - BUBA - Frambesia - Parangi - Paru - be enlarge certifugilly - symmetrically on anterior wrists, dorsal hand, small joints over hand or feet	- Penciline, but treatment in the infection stage prevent the characteristic late leukoderma

بسم الله الرحمن الرحيم

قَالوُاْ سُبْحَانَكَ لاعِلْمَ لَنَآ إلا مَا عَلمْتَنَآ إنكَ أنْتَ العَلِيمُ الحَكِيمُ

صدق الله العلي العظيم

البقرة- ٣٢

Supervisor Certification

We certify that this thesis was prepared under our supervision at the Department of Computer Science / College of Science / Al-Nahrain University, by **Boraq Fadel Neema Al- Shannoon** as a partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Signature:

Name: **Ban Nadeem Thannoon**

Title: **Assist Proff.**

Date: / /2004

In view of the available recommendations, I forward this thesis for debate by the examination committee.

Signature:

Name: **Dr. Taha S. Bashaga**

Title: Head of the Department of Computer Science, Al-Nahrain University.

Date: / / 2004

Certification of the Examination Committee

We chairman and members of the examination committee, certify that we have studied the thesis entitled **(Development Of An Expert System For The Diagnoses Of Skin Diseases)** presented by the student **Boraq Fadel Neema Al- Shannoon** and examined him its content and in what is related to it, and we have found it worthy to be accepted for the degree of Master of Science in Computer Science.

Signature:
Name: Dr. Imad Hussain Marza
Title: Assistant Professor
Date: / /2004
(Chairman)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature: Dr.Ban Nadeem Thannoon
Name:
Title: **Lecturer**
Date: **/ /2004**
(Supervisor)

Signature:
Name: **Laith Al Anne**
Title: **Dean of Collage of Science**
Date: **/ /2004**

Dedication

To my beloved family

Acknowledgments

I would like to express my sincere gratitude and appreciation to my supervisors **Dr. Ban Nadeem Thannoon** and **Dr. Rabab Saady** for their valuable guidance, supervision and their efforts during the development of this study.

Grateful thanks for the Department of Computer Science's stuff and employees for their support during study.

Special thanks to my friends and family for their continuous support and encouragement during the period of my study.

CHAPTER ONE

INTRODUCTION

1.1 Introduction

Expert systems are computer programs that are derived from a branch of computer science research called Artificial Intelligence (AI).

AI is the study of how to make computer do things, which at the moment; people do better [Ger01].

Expert systems combine a knowledge base of rule and domain-specific facts. They have architecture that separates the knowledge base from the inference engine. This feature make the knowledge base more easily modified as new rules and facts become known. Reasoning in expert systems can be explained. Most expert systems allow the creation of explanation systems to help the user understand question being asked or conclusions being reached.

There have been various application areas that have been successful for expert system development. Application areas include classification, prediction, diagnosis, planning, monitoring, scheduling, maintenance and targeting [SRI01].

To perform difficult tasks, we need to focus our knowledge on narrow range of applications. This will be helpful for enabling to explain their actions and lines of reasoning.

ESs are fit where there are no established hypothesis , where the information is ‘cloudy’ or ‘fuzzy’ and where human expertise is scare or in high demand and their services are expensive.

1.2 Literature Survey

Several research efforts that are concerned with developing and designing expert system have been published. Some of these efforts are summarized below:

1. [STE92], In that orientation a prototype expert system named Recurrent Abdominal Pain System (RAPS) has been implemented successfully for medical diagnosis. It is a rule-based system use a backward chaining approach and employs a certainty factor method for its reasoning. Other features of RAPS include good explanation facilities and interact with the user to ease the use. It also allows the system to deal with knowledge base that was acquired from expertise.

2. [THA94], in this orientation, an expert system application in Iraqi law (ESAIL) attempts to provide means to manipulate an Iraqi law domain, which is one of the most popular areas in Iraqi's real life word. The system explains its conclusions by applying the law text associated with. ESAIL is a rule-based system, which store its knowledge in an external database and its uses the B^+-tree indexing method for retrieval from the database. In its present form, ESAIL deal with restricted domain in the laws related to the Iraqi personal status law, with its various sections. ESAIL deals with two types of users: the professional lawyers who have unrestricted access to the system during the input of new laws and modifications by interactive procedures; while ordinary users have access only to the use of the system, without being able to enter or modify laws.

3. [KAN98], this work is an attempt to develop a conceptual scheme for rule-based connectionist expert system. A new approach is proposed to integrate neural networks with rule-based expert system. The integrated system combines the strength of rule-based semantic structure and the

learning capability of connectionist architecture. The developed structure had used the subsystems of neural networks (subnets) as the basic building blocks, where the subnet represents a cluster of rules. These subnets were organized into a hierarchical structure that corresponds to the deduction steps of the expert system. The scheme also, considers reasoning under uncertainty. The aim is to communicate a confidence factor to the user to give some guidance to the validity of the deducted facts

4. [SRI01], Regular of Mean Arterial Pressure (MAP) using Sodium Nitro Prusside (SNP) infusion is common in many hospitals. This research comparatively evaluated the performance of three types of expert system controllers to automate this task: rule-based, fuzzy, and artificial neural network. For meaningful comparisons the three systems were based on the same set of rules. Their performance was tested on a nonlinear blood pressure model derived and scaled from canine data that simulate typical patient responses to the drug. The controllers were tested for different patient sensitivities, baseline drift, and noisy blood pressure readings the controllers were able to regulate the MAP in the target region about the set point for more than 90% of the time. The rule base controller reduced MAPS the faster, while the fuzzy and neural Controllers regulated MAP better over longer periods.

5. [NIE01] , detailed design assumptions and functional principles of two hybrid rule-and model-based expert system shells for uncertain backward and forward reasoning were presented. The systems may used to reason with any knowledge base, The knowledge Base semantics is simple, intuitive and straightforward. Practically unlimited nesting of rules, methods as well as rules and models was allowed. Uncertainty was modeled using a modified

Stanford Certainty Factor Algebra. The classic Stanford Certainty Factor Algebra using Certainty Factors in the range [-1,1] has been modified to make it conform to Aristotelian Logic for the case of CF's being equal 1 or -1 and take into account dependant and independent lists of conditions. The system were equipped with diagnostic facilities which can automatically checking the rule and constraint base for inconsistencies and redundancies, providing warnings and detailed diagnostic messages.

6. [ALK02], this work was an attempt of designing an expert system shell for general diagnosing systems that contains all the facilities to help its users (human expert) to complete their jobs (build their own expert system) easily and efficiency. Two shells were designed and implemented. The first was called **GADES**, which stands for General Automatic Diagnosing Expert System, While the second was called **GACES,** which stands for General Automatic Code Expert System. Both designed shells were constructed as two phases shell. These two phases are: an automatic friendly inference (expert interface), and an inference engine. The first face is responsible for acquiring human expert knowledge in an automatic way by using menus and message windows. The second phase is concerned with the design of inference engine together with a user interface that uses a backward chaining. The two designed shells based on reasoning under uncertainty using certainty factor method (Stanford Certainty Factor).

1.3 Aim of the project

This research is an attempt to develop an expert system to diagnose the hypomelanosis skin diseases with their treatments, which is based on an uncertainty.

1.4 Thesis layout

The following are the layout of the thesis contents:

- **Chapter one** Includes an introduction to Expert Systems and literature survey to the related work done on it.
- **Chapter two** contains theoretical concept of expert system , this include what expert systems are and the characteristics of an expert system , essential elements of it , knowledge representation in general , knowledge acquisition methods, reasoning under uncertainty, and some application in expert system.
- **Chapter three** contains the development of the suggested system.
- **Chapter four** includes the operations of the system with the user interface.
- **Chapter five** include the discussion, conclusions of this work are given together with some recommendation for future work in this field.

CHAPTER TWO

THEORETICAL CONCEPTS OF EXPERT SYSTEMS

2.1 Introduction

Expert System (ES) can be defined as a program that attempts to mimic human expertise by applying inference methods to a specific body of knowledge (Domain) [DAR00].

Such system would fulfill any function through human expertise, or it could be of assistance to human decision maker. The decision maker may be qualified, in such case the program prove its existence by the decision maker productivity. In other way, the human collaborator may be capable of reaching expert levels of performance using certain technical help from the program [STE92].

Before exploring expert system, it is important to be clear about the distinction between three concepts: data, information and knowledge, at which the difference between them is the question of levels. **Data** is a group of alphabetic symbols (just uninterrupted values, e.g. (46), **information** is passive in the sense that it does not give rise to further generations of information ("organized values", which can be regarded as having some sense or interpretation,e.g 46 held as the 'age' field in a personal details' record), while **knowledge** is active in that it can give rise to further generations of information (information which is known to be true),which represents the main difference between knowledge and information [DAR00].

In this chapter we will discuss the following subjects

1. Expert System Characteristics.
2. Expert System Components.

3. Knowledge base representation
4. Knowledge Acquisition.
5. Reasoning under uncertainty
6. Some applications in expert systems

2.2 Expert Systems Characteristics

Expert Systems have many characteristics, which can be summarized as follows:

1. It simulates human interpretation of problem domain, rather than simulating expert systems from more familiar programs that involve mathematical modeling. This do not mean that the program is ideal psychological model of the expert, but it focuses on emulating of expert’s problem solving abilities, Thus performing the relevant tasks as an expert or better.
2. It solve problem by heuristic or approximate methods, which unlike solutions, are not guaranteed to succeed. A heuristic is essentially a rule of thumb that implies the expression of a piece of knowledge about how to solve problems in some domain. Such methods are approximate in the sense that they don’t require precise data and the solutions derived by the system may be proposed with varying degrees of certainty [STE92].
3. Good reliability. The expert system must be reliable and not prone to crashes or it will not be used [LUG89].
4. The ability to dealing with uncertainty and incompleteness. Information about the problem to be solved can be incomplete or unreliable, relation in problem domain can be approximate, for example we may not quite sure that some symptoms are present in the patient, or the measurement data

is absolutely correct, some drug may cause some problem, but usually does not. All this required reasoning with certainty [BRA01].

2.3 Expert System Components

Most of the ES are composed of a user interface, knowledge base, an inference engine, and methods for building and updating the knowledge base. Figure (2.1) is an illustration of expert system architecture [TUR98].

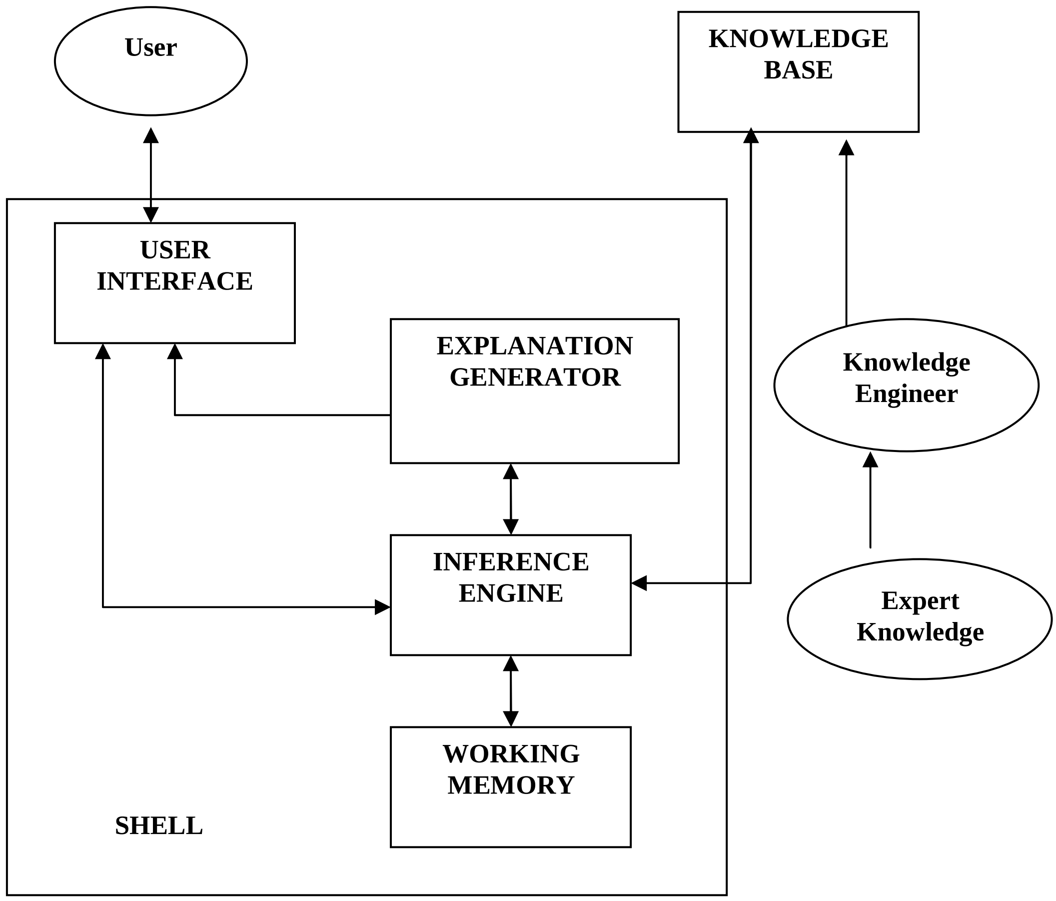


Figure (2.1) Components of an Expert System

2.3.1 User Interface [STE92]

The interfaces permit the user to interact with the system to present the problem and find the conclusion. The system can justify its conclusion, as well as human expert can inform why particular option were taken or ignored.

The user interface tries to prove similar form of communication facilities provided by the expert , but interface has limited capability for understanding natural languages and general world knowledge, however sometimes graphic user interface can make available a form of human-human communication that has no direct analog in human-human communication .

2.3.2 Knowledge Base

The heart of the expert system is the general knowledge base, which considers the problem solving knowledge of the particular application. It contains both general knowledge as well as a case specific knowledge [LUG89].

A knowledge base comprises the knowledge that is specific to the domain of application, including such things as simple facts about the domain, rules that describe relations or phenomena in the domain, and possibly also methods, heuristic and ideas for solving problems in this domain [BRA01].

There are standard set of knowledge representation techniques, any of which can be used, each techniques provide the program with certain benefits such as making it more efficient, more easily understood or more easily modified.

The three most widely used in current expert systems are rules, semantic net, and frame [WAT86].

2.3.3 Inference Engine

The inference engine applies the knowledge (in knowledge base) to the solution of actual problems. It is essentially an interpreter for the knowledge base. In the production system, the inference engine performs the recognize-act control cycle. The procedure that implements the control cycle is separated from the production rules themselves [LUG98], [WAT86].

An important aspect of the inference engine is the kind of control of guidance that is available to direct the search for an answer. Such control mechanisms are referred to as search strategies that are responsible for determining the orders in which rules are selected. Among the most widely used control strategies are forward and backward reasoning (or chaining) [BRA01], [WAT86].

Both control strategies are described as below:

In data driven search, sometimes called **forward chaining,** the problem solver begins with the given facts of the problem and a set of legal moves or rules for chaining state. Search proceeds by applying rules to facts to produce new facts, which are in turn used by the rules to generate more new facts. This process continues until it generates a path that satisfies the goal condition.

In goal driven search, sometimes called **backward chaining**, take the goal that we want to achieve, see what rules or legal moves could be used to generate this goal and determined what conditions must be true to use them. These conditions become the new goals, or sub goals, for the search. Search continues, working backward through successive sub goals until it works back to the fact of the problem. This finds the chain of moves or rules leading from data to a goal, although it does so in backward order [LUG89].

In rule-based expert systems, the inference engine works by selecting a rule for testing and then checking if the conditions for that rule are fired. The conditions may be found by questioning the user or they may be already discovered facts during the consultation when the conditions of the rule are found to be true, then the conclusion of the rule is true. The rule is then said to have “fired”. The conclusion of this rule will then be added to the knowledge base or may be displayed via the user interface for information [RIC91].

The important feature of inference is the ability to deal with problems whenever the knowledge is uncertain. This would help during diagnosis if the data or evidence for the cause of a fault is uncertain. The user will typically express this as a degree of certainty or probability for the occurrence of the evidence [PAR88].

Three inference strategies commonly used by experts are: deductive inference, inductive inference and abductive inference. These will describe in some detail [ALK02].

- ***Deductive inference*:** The attraction of deductive inference is that it is a form of reasoning that is mathematically exact. This means that if the premises are true, then the conclusion is guaranteed also to be true.
- ***Inductive inference*:** It is a form of reasoning from specific to general. This form of inference lacks the mathematical exactness of deduction meaning that there is always the possibility that the conclusions are false. However, inductive inference is common inexpert systems because it does match human inference in the real world. To prove a conclusion, some additional assumption would have to be included from human common-sense knowledge of the world of the world and added to the given premise.

- ***Abductive reasoning***: Abductive inference explains effects in terms of their cause. This contrasts with deductive inference, which works from causes to effects. For example, consider the rule "if it is training then the grass in the garden will get wet".The truth of the premise will ensure the truth of the conclusion. This is "cause and effect". Abductive reasoning**,** when applied to this rule**,** would assume the truth of converse. That is "if the grass in the garden is wet then it has been raining". This rule would be generally accepted even though there is no guarantee of its correctness. It is possible that the grass has become wet because it has been sprayed with water, or perhaps for some other reason. Thus, abductive reasoning also lacks the mathematical exactness of deductive reasoning. However, human experts frequently apply abductive inference.

2.3.4 Working Memory

The working memory is a dynamic image of the knowledge (which is usually empty at any initial operation time). It consists of facts that acquired by the expert system during its operation with dynamic data feeding.

Furthermore, it records the intermediate results and hypotheses that expert system manipulates.

As for operation progress, the inference engine uses fact and rules in the rule base, in conjunctions with user input, to add facts to the working memory [TOW88].

The working memory can be finite or “unlimited” in size. Various strategies can be used in the finite case when the working memory fills up, for example drop the oldest data, or keep only the important data [TOB91].

2.3.5 Explanation Facility

Confidence in a computer system achieved not only from the quality of its results but also from the assurance that the system's reasoning is perfect and in coincidence with the task required [PAR88].

Majority of the current expert systems posses an "explanation facility" which is knowledge for explaining "How" the system works and "Why" it's actions or result are appropriate [WAT86]].

The term **"Why"** under "explanation facility" gives answer for question about why some conclusions were reached or why some alternatives were rejected. To carry out this task, the system uses a few general types of question answering plans, and this requires the system to trace backward along working memory from the current goal toward the top goal.

The system can also explain **"How"** it has arrived to a certain conclusion by tracing forward, along working memory from the top goal toward the current goal. Also it can work backward.

An inference system that can explain its behavior on demand will seem much more believable and intelligent to its users [MAR89].

2.3.6 Shells [DAR00]

Shell provides an easy starting point for building an expert system because of their ease of use. They are expert systems that have been emptied of their rules. This means that developers can concentrate on entering the knowledge base without having to build every thing, including the inference engine and user interface, from scratch. Even non-programming experts can familiarize themselves with shells fairly rapidly. Many expert system shells contain facilities that can simplify knowledge acquisition. Non-programming experts can acquire an understanding of shells without undertaking the lengthy learning process that programming other types of software development requires. However, using a shell to build an expert system can lead the builder into oversimplifying the application domain because shells are inflexible, in that it is difficult to modify or change the way they work with regard to both representation of knowledge and the inference mechanism.

2.4 Knowledge Based Representation

The knowledge concerning any problem domain is absolutely necessary to solve problems. If computers are to solve problems, then the knowledge must be encoded into data structures that can be created and used by program [STE92].

The techniques used to represent knowledge in expert system application are: - [STE92] [DAR00]

1- Rule based knowledge representation.
2- Semantic network knowledge representation.
3- Frame based knowledge representation.
4- Formal logic knowledge representation.

2.4.1 Rule Based Knowledge Representation

The majority of expert system use rules sometimes called productions, to represent knowledge [DAR00].

A rule consists of two parts: condition (antecedent) part and conclusion (action, consequent) part i.e.:

If (conditions) then (conclusion)

Antecedent part of the rule describes the facts or conditions that must exist for the rule to fire consequent describes the fact that will be established or the action that will be taken or conclusion that will be made [web1], for example:

If

Animal is bird and

It does not fly and

It swims and

It black and white

Then

It is a penguin

Following this model, an expert system will receive propositions, or answers to a certain line of questions. It will compare the propositions with the facts and rules registered in its knowledge base. Using the inference engine, it will evaluate the propositions against the rules and infer an answer.

The piece of knowledge represented by the production rule is relevant to the line of reasoning being developed if the **If** part of the rule is satisfied; consequently, the **Then** part can be concluded, or it's problem-solving action taken. A rule may fire another rule, which in turn may seem more information from the user [Ger01].

Some rules may also be true but not always all the time; i.e., they may be uncertain or inexact. The way of expressing uncertainty about the rule is by adding a certainty factor (CF) to the rule, which indicates the degree of the rule certainty being true [WEI88].

A rule based representation has the following advantages [DAR00]:

a- Simplicity. Rule form a good psychological model for knowledge representation because they closely relate to human reasoning. This makes rule based system easy to build in comparison with other methods for representing knowledge.

b- Modularity. This means that blocks of rules can be independently written and added to a rule base, and checked for correctness. So rule based expert systems can be broken down into easily manageable components for development. This feature that enables knowledge bases to be constructed incrementally, step by step.

c- Handling uncertainty. A number of technique have been developed that allow knowledge about uncertainty to be contained within rules.

2.4.2 Semantic Network Knowledge Representation

A semantic network, or net, is a classic representation technique used for prepositional net.

The structure of a semantic net is shown graphically in term of nodes and the arcs connecting them. Node are often referred as objects and the arcs as links or edges.

The links of semantic net are used to express relationship. Nodes are generally used to represent physical object, concept, or situation [JOS98]. Relatinship are of primary importance in semantic nets because they provide the basic structure for organizing knowledge. Without relationship, knowledge is simply a collection of unrelated fact, with relationship, knowledge is cohesive structure about which other knowledge can be inferred [JOS98].

The problem with semantic net is the dificulty of updating them to reflect new knowledge or changed relationships.[STE92][ALK02].

2.4.3 Frame Based Knowledge Representation

Frames representation combines the idea of semantic networks and rules. A frame is a complex data structure that collects together knowledge about a particular concept and provides expectations and default knowledge about the concept. Typically, the frame is represented as a template, which consists of a number of slots and optionally the values that the slots can take. The values may themselves be other frames. A new concept can often be represented by merely adding a frame for that concept. Missing information about a concept is readily apparent, the slot for that information is empty. Prior filling of the slots can provide default information easily [BRA01],[TUR98].

The expressive power of simple frame systems found in two features of the representation: -

a. The frame contains descriptive details of an object in question and it can embody procedural knowledge besides declarative proprieties and value. This is executed by procedural attachments, depending on the state of a particular slot and procedures can be called to execute necessary computations.
b. Information can be transferred within frames, using an inheritance link. Using such a link, the lower frames in a hierarchy can inherit values in their slots from an upper frame [FOR89].

2.4.4 Formal Logic Knowledge Representation

This was first followed-up by the ancient Greek philosophers in search for understanding reasoning and knowledge, involving the mathematical and philosophical study of logic. The logic is based on the concept of truth because in formal logic, a statement is either true or false [WEI88].

There are two primary examples of this type prepositional logic and predicate logic [WEI88].

a. *propositional logic:* Propositional logic covers the knowledge that can be represented by using a series of statement called proposition [RIC91].

Propositional simply denote propositions, or statement about the world that may be either true or false such as: [LUG98],[LUG89].

(1) “The car is red”

(2) “Waters is wet”

The most common connectives used in prepositional logic are shown in table (2.1) and table (2.2) summarizes the advantages and disadvantages of the three representation scheme described above , and table (2.3) summarize the main characteristics of the three representation schemes [DAR00].

Table (2.1) Connectives used in prepositional logic

Symbol	Meanings	Interpretation
~A	Not A	Negation, Negation of proposition A is true if A is false and vice versa.
A ^ B	*A And B*	Conjunction. A and B only true if A and B are both true, otherwise false.
A v B	*A Or B*	Disjunction. A or B is true if A is true or B is true.
$A \rightarrow B$	*A Implies B*	Implication. If A is true and A implies B is true, then B is true. If A is false and A implies B is true then anything goes. That is, B could be true or false, since implication says nothing about case when A is false.

b. Predicate logic: Predicate logic is much more than flexible prepositional logic since it allows the use of predicates and quantifiers within the logic statement [STE92].

It is inadequate for solving some problems because a proposition has to be treated as a single entity that is either true or false. Predicate logic overcomes this by allowing a preposition to be broken down into two components. These are known as arguments and predicates [DAR00].
Predicate calculus also allows expressions to contain variables. Variables let us create general assertions about classes of entities. For example, we could state that
For all values of X, where X is the day of week, the statement:-
Weather (X,rain) is true; i.e., it is rain every day [LUG98], [LUG89] .

2.5 Knowledge Acquisition [PAR88]

"Knowledge acquisition is the transfer and transformation of problem-solving expertise from some knowledge source to a program" and such knowledge source derived from human experts, textbooks, databases and a person's own experience.

The transfer and transformation required to offer expertise for a program could be automated or partially automated in some special cases. A second person (analyst or knowledge engineer) will be required to communicate with expert and the program.

A number of approaches to knowledge acquisition have been suggested. The three basic approaches are as follows:-

a-Interview. Such approach implemented against knowledge passed to a knowledge engineer for human expert through many interviews and by encoding such knowledge in the expert system in this case, the knowledge engineer obtains the heavy burden in the knowledge acquisition process, and the expert system quality entirely rely on the knowledge engineer's skill.

Table (2.2) Advantages and disadvantages of representation schemes:

Representation	Advantages	Disadvantage
Production rule	• modular, • flexible • well suited to many domain	• Difficult to represent descriptive knowledge in natural way • Difficult to separate domain knowledge and problem-solving knowledge.
Semantic network	• Object based representation therefore, permits inheritance	• Can not distinguish between the class of an object and a particular object. • Unlike frame, no facility to handle procedural knowledge. • Presentation and structure for complex systems could become unmanageable.
Frame	• Object-based representation. • Facilities reusability. • Frames offer facilities for exception handing and defaults, both of which are not easily handled in logic or other representation.	• The oritical difficulties arise from slots in frame being unrestrained.this mean nothing can be certain and so it can be impossible to give universal definitions to objects.
Logic	• Precision that is, conclusions guaranteed to be correct if premises correct. • Allows programs to be decleritive,like prolog.	• Opaque, poor psychological model. • Difficult to represent uncertainty.

Table (2.3) Main characteristics of representation schemes

Representation	Basic architecture	Method(s) of inference
Production rules	• Consist of set of rules, database of known facts and interpreters, system work by applying known fact to left-hand side of rule; if true, right hand-side fires. Newly discovered fact is added to database. Order in which rules are selected depends on the method of inference.	Forward and backward chaining. In forward chaining, the interpreter searches the data to see which fact match the left hand side of the rule. In backward chaining the interpreter tries to prove goal by attempting to confirm the conditions leading to the goal.
Semantic net	• Object based representation uses link and nodes to represent associative knowledge.	Inheritance through the links and nodes
Frames	• Object-based representation uses slots for storing attributes. • Default value can be assigned to solve values, thus facilitating inheritance.	• Inheritance through frame hierarchy. Slot value can be set as default use local slot value to override any parent slot value.
Logic	• Uses clauses, rules of inference such as modus pones and system output in form of questions.	Resolution

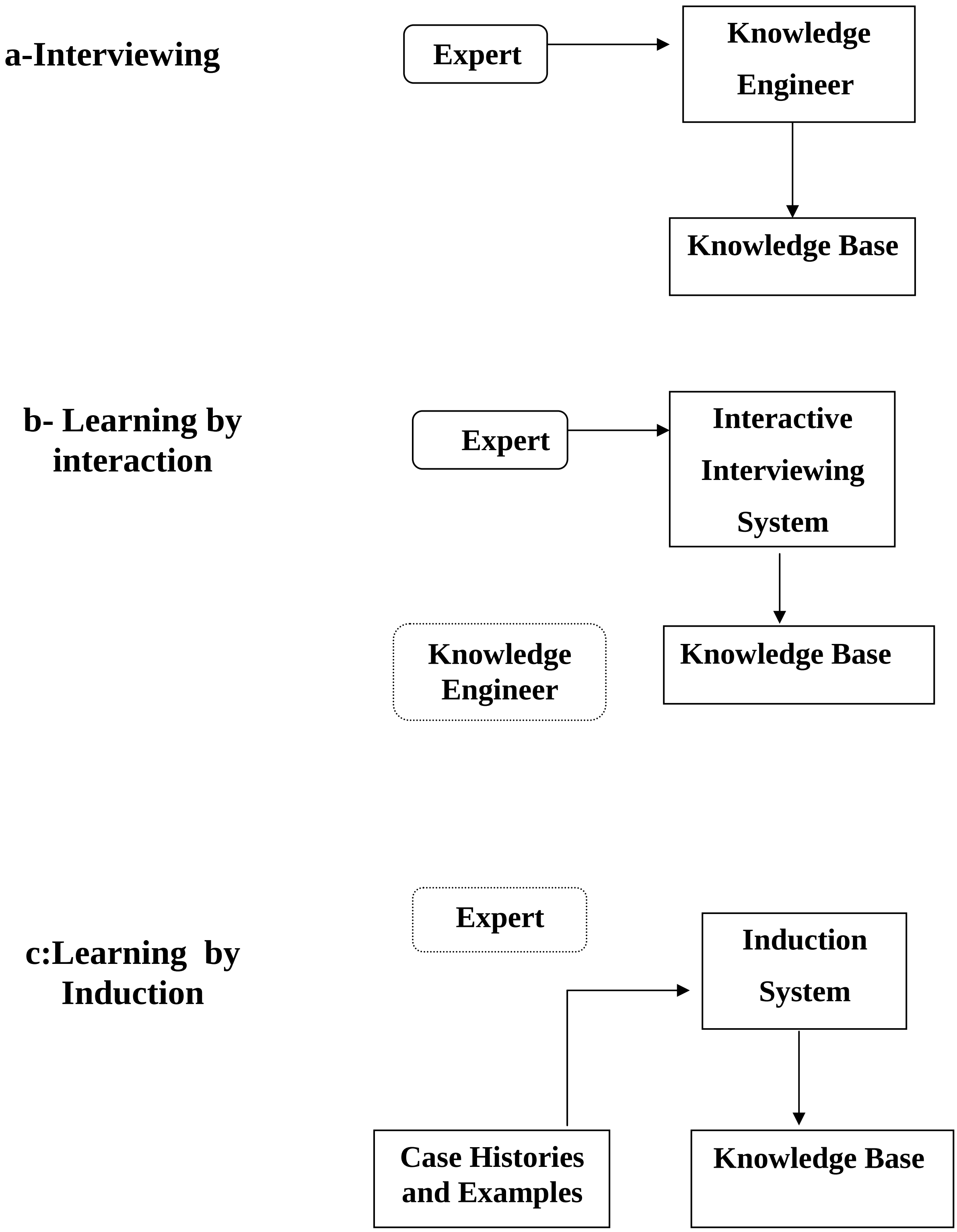
<u>b- Learning by interaction</u>. This approach will depend on computer-assisted knowledge acquisition. Here experts directly interact with a computer program that helps to capture their knowledge; a knowledge engineer role almost vanished and the program often helps experts to clarify their own thoughts.

<u>c- learning by induction</u>. In this approach a computer program makes available the best knowledge by examining data and example. Here the reliance on both expert and the knowledge engineer is again diminished. The main problem here is how to identify the most suitable characteristics or attributes on which induction would be performed properly.

Interviewing is the most desirable adopted method of knowledge acquisistion, although, it mainly depends on the knowledge engineer and is time-consuming and expensive.

“Approach **b** focuses on specific interaction methodologies and interactive interviewing techniques that help expert discover the structure of this own knowledge knowledge in selected tasks.

Approach **c** focuses on algorithms that analyze data and examples and then generalize them to obtain the required knowledge". Figure (2.2), shows the three approaches of knowledge acquisition:-



Figure(2.2) Approaches to knowledge acquisition

2.6 Reasoning Under Uncertainty in Expert System [JOS98]

Uncertainty can be considered as the lack of adequate information to make a decision. It is a problem because it may prevent us from making the best decision and may even cause a bad decision to be made. In medicine uncertainty may overlook the best treatment for a patient or contribute to an incorrect therapy. In business uncertainty may mean financial loose instead of profit. When uncertainty facts are involved, the number of possible conclusions may greatly increase the difficulty in finding the best conclusion.

Although there are many of expert systems applications that can be done with exact reasoning, many others required inexact reasoning involving uncertain facts, rule or both. Classic examples of successful expert systems that deal with uncertainty are MYCIN for medical diagnosis.

Unfortunately, determining the best conclusion may not be easy. A number of different methods have been proposed for dealing with uncertainty and aid in choosing the best conclusion.

2.6.1 Methods of handling uncertainty [web3]

Different approaches have been proposed to handling uncertainty in expert system the principal ones are:-

- Bayesian probability.
- Stanford certainty factor (CF).
- Dempstar-Shafer (DS) theory of evidence.
- Fuzzy logic.

2.6.1.1. Bayesian Probability.

An old but still very important tool in AI problem solving is probability. Probability is a quantitative way of dealing with uncertainty that organized in the seventeenth century [JOS98]. It is the chance that particular events will occurs (or not occurs). It is ratio computed as follows:

$$p(x) = \frac{\text{Number of outcomes favoring occurrence of x}}{\text{Total number of outcomes}} \quad \ldots\ldots\ldots\ldots(2.1)$$

The probabilities of x occurring , stated as P(x), is the ratio of the number of times x occurs to the total number of events that take place. One of the most important results of probability theory, the general form of Bayes' theorem. Bays provides a way of computing the probability of a hypothesis Hi, following from a particular piece of evidence, given only the probabilities with which the evidence follows from actual causes (hypothesis).

$$p(H_i / E) = \frac{\mathrm{P(E / H_i) * P(H_i)}}{\sum_{k=1}^{n} \mathrm{P\,(E\,/H_k) * P\,(H_k)}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.2)$$

Where:

P (H_i/E) is the probability that H_i is true given evidence E.

P (H_i) is the probability that Hi is true overall.

P (E/H_i) is the probability of observing evidence E where H_i is true

n is the number of possible hypotheses.

There are two major requirements for the use of Bayes' theorem: First all the probabilities on the relationships of the evidence with the various hypotheses must be known, as well as the probabilistic relationships among the pieces of evidence. Second, and sometimes more difficult to establish, is that all relationships between evidence and hypotheses, or

P (E/H_k), must be independent. In general, and especially in area such as medicine, this assumption of independence cannot be justified.

A final problem, which again makes keeping the statistics of complex Bayesian system intractable, is the need to rebuild probability tables when new relationships between hypotheses and evidence are discovered. In many active research areas such as medicine, new discoveries happen continuously. Bayesian reasoning requires complete and up-to-date probabilities, including joint probabilities, if its conclusions are to be correct. In many domains, such extensive data collection and verification are not possible.

Where these assumptions are met, Bayesian approaches offer the benefit of a mathematically founded handling of uncertainty. Most expert system domains don't meet these requirements and must rely on heuristic approaches LUG98].

2.6.1.2. Stanford Certainty Factor [web4].

The certainty factor CF was modeled as a difference between two positive measures, MB-Measure of Belief, and MD-Measure of disbelief, both in the range of [0, 1].

CF= MB-MD.......................................(2.3)

Each of these was manipulated separately by similar rules as for the CF value, and the conclusions thus were represented as intervals of beliefs rather than single numbers. Thus, a single CF number, e.g. 0 may present total ignorance or the difference between equally strong but opposite beliefs.

The best known of the nonprobablistic method s is a method used by a diagnostic expert system MYCIN. In MYCIN, sentences are assessed not with probabilities (which would range from 0 to 1) but with certainty factors which range from –1 to +1. A certainty factor of –1 indicates that the sentence in question known to be false; a certainty factor of +1 indicate that it is known to be true. A certainty factor of 0 indicates no belief either way.

There are three manners in which the certainty factors are combined during the evaluation.

a- Antecedent Combination Rule

When reasoning about a particular query Q, MYCIN may decide for a rule that says

If P1 and. . . . and Pn then Q .

In order to do so, however, the system needs to assign a certainty factor to the conjunction P1 and . . . Pn; the value given by MYCIN for this expression is simply the minimum of the certainty factors assigned to each of the Pi's. the intuitive justification of that is that we can not be more convinced of the conjunction that we are of any of the conjuncts it contains and we certainly get the right answers in the case where each of the Pi's is true or where one of them is false.

Similarly, disjunction ('or') is computed by the maximum values, while logical negation('not') is computed by negation.

b-Serial combination rule

The results of applying the rules are chained, using a special version of modus ponens that is appropriate for use with certainty factors.

If a sentence (conjunction) P has a computed certainty factor C and the rule is:

If P then D certainty Q

Then the certainty factor assigned to Q is given by

C * D	if C >0	where C is the Certainty Factor
0	Otherwise.	

If C is non-positive, the premises for the rules are not true, so the rule is irrelevant, and beliefs should therefore be unaffected.

c-Paralled combination rule

If a conclusion Q is implied by two (or more) rules like in the example

If P1 then . . . certainty Q

If P2 then . . . certainty Q

And Ci is the certainty computed by rule i, then the certainty C for Q is computed according to the formula:

C = C1 + C2 - C1 * C2 if C1, C2 > 0 ….… (2.4)

C = C1 + C2 + C1 * C2 if C1, C2 < 0……….(2.5)

C=(C1 + C2) / (1- min(abs (C1) , abs (C2)) if C1 * C2 <0 ….(2.6)

2.6.1.3 Dempster-Shafer Theory [web2]

An alternative to Bayesian network is a dempster Shafer theory, which is design to deal directly with the distinction between unceretainty and ignorance. It computes the probabilities that the evidence supports the propositions. This measure of belief is called belief function, written Bel(X), rather than computing the probability of propositions.

To define the belief Bel(X) precisely, there is a need to start with a full list of mutually exclusive hypotheses. This hypothesis space is called the frame of discrement, denoted by θ. In simplified medical diagnosis system, θ could be {allergy, cold, flu, pneumona}. The goal is to attach some mesure of belief to each element of θ.

Note that evidence often supports more than one hypothesis. Also, because the hypotheses are mutually exclusive, evidence in favor of one hypothesis may affect the belief of another. In a purely Bayesian system, handling this is done by listing all the combinations of conditional probabilities. Dempster-Shafer theory attempts to avoid the need for that by manipulating the sets of hypotheses directly.

If θ has n elements, it will have 2^n sub-sets and the basic probability assignment m(X) measures the amount of belief currently assigned to each sub-set X. by assigning these so that the sum of m(X) over all sub-sets $X \subseteq \theta$ is 1 . Corresponding sets have no relevance to the problem.

2.6.1.4 Fuzzy Set Theory [web2]

Another popular alternative theory for uncertain reasoning in expert system is fuzzy logic.This is based on the idea that many concepts are not sharply defined (e.g. fast, tall, hot) and consequently, standard set theory and the if-then rules cannot be used when reasoning with them. Fuzzy logic is built upon the underlying idea of fuzzy set theory.

Classical set theory is based on two valued logic – relations such as $X \in S$ are either true or false. Such classical sets are sometimes called **crisp** sets. For example the definition of a crisp set of fast cars, which have a top speed greater than 150 mph:

FastCars= $\{X \in \text{Cars} : \text{TopSpeed}(X) > 150 \text{ mph}\}$

But the concept of fast car is not really precise like that. It is more reasonable to define it as a **fuzzy set** with elements that are members to a certain degree. A fuzzy set is defined as a function from the appropriate domain to the interval **[0, 1]** such that f(X) = 1 denotes X is definitely a member,f(x) = 0 denotes x is definitely not a member , and other values denoteintermediate degrees of membership.

Just as classical set theory is governed by two- valued logic, fuzzy set theory can be related to a many- valued logic in which propositions such as Fast Cars (X) have values in the interval [0,1] and extensions to the standard rules of logic can be defined.

The negation of fuzzy predicate f{X} can be defined as in probability theory:

$\neg$ f(X) = 1- f(X) ……………………………(2.7)

the rules for evaluating the fuzzy truth of other operators are less obvious:

f(A $\wedge$ B)= min(f(A),f(B))………………….. (2.8)

f(A $\vee$ B)= max(f(A),f(B))…………………. (2.9)

f(A $\Rightarrow$ B) = min (1,1-f(A)+f(B))……………(2.10)

these are similar to those used with the certainty factors in MYCIN. Note that these imply f (A $\vee$ $\neg$ A) $\neq$ f(True) , which might be expected to lead to problems.

Fuzzy predicates and fuzzy logic could be applied to the rules (of the kind used in expert systems) in a similar manner to MYCIN style certainty factor. Fuzzy logic operates could be used to compute the degree of truth of the condition and take that to be the degree of truth of the conditions and take that to be the degree of truth of the action. If the rule itself is only true to a certain degree, then we have to factor that in as well.

Building fuzzy experts follows the same procedures as any other expert system, except that getting the expert system to define all the fuzzy rules, and make sure they are all internally consistent.

Fuzzy logic based expert systems have been very successful in commercial applications, but these have been rather small, with limited levels of inference, and parameters tuned by machine learning. Various counter intuitive features, such as $f(A \vee \neg A) \neq f(\text{True})$,have led some people to regard the use of fuzzy logic in more complex expert systems to be as problematic and unreliable as the use of MYCIN style certainty factors.

2.7 Some Applications in Expert system

2.7.1 MYCIN Expert System

The MYCIN project was a cooperative venture by the Department of Computer Science and the Medical School at Stanford University. The main work on project was done during the middle and late 1970s, and about 50 person-years were expended in the effort. MYCIN was designed to solve the problem of diagnosing and recommending treatment for meningitis and bacteremia (blood infection) [LUG89].

The name of MYCIN was taken from the ordinary suffix shared by distinct drugs used in treatment, example, Erythromycin and Ampicillin. The main intention of the system was to provide advice, and it does so via the consultation program figure (2.3). As well as to this, three programs that increase system utility and flexibility were adjusted:

a- Question-answering program that manipulates languages inquires about either explicit consultation or the general knowledge base of the program.

b- An explination program for giving reasons why the performance program has demanded an item of information or how it drown particular conclusion. It allows the use of the term **WHY** and **HOW** for this purpose.

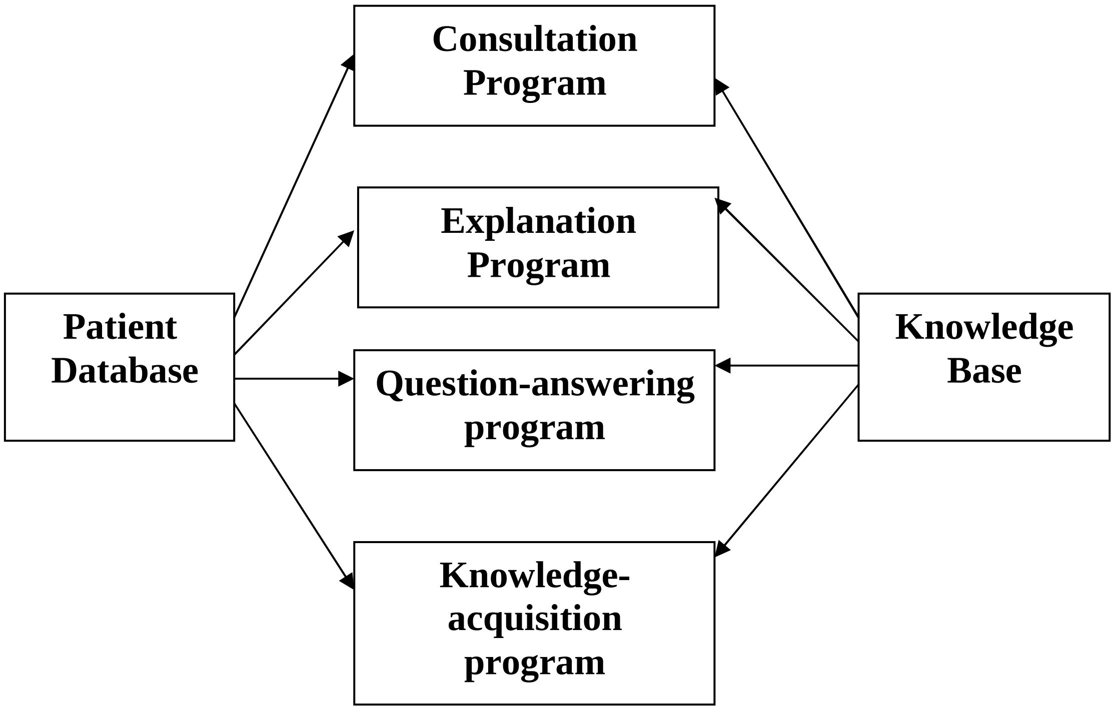


Figure (2.3) Component of MYCIN system

c- A knowledge acquisition program allowed for extending the knowledge base by means of a dialog with an expert.

MYCIN is a rule-based system consist of handereds of rules such as the following :

RULE NO.50

IF (1) the infection is primary bacteremia ,and

(2) the site of the culture is a sterile site,and

(3) the suspected portal of entry of the organism is gastro-intenstinalt tract

THEN there is suggestive evidence (0.7) that the identity of the organism is bactereroidws.

MYCIN use backward chaining and depth-first search through AND/OR graph and is coded in INTERLISP programming [THA94].

Some important features of MYCIN are as follows:

1- The program can produce a result (in the case a diagnosis) based on facts supplied by the user, the facts stored in its database and its rule of inference. These facts are represented as triples in the form (context – parameter – value) and also associated with certainty factor number ,as in the following example:
(Patient- age- 43 0.8).

2- It can allocate a probability on a scale between 1 for "certain", thorough 0 for "don’t know " to -1 for " certainty not".

3- It can be generalized to cover areas other than infection.

The disadvantages of **MYCIN** are:

* It requires larg amount of typing, with a sole interview taking 30 to 40 minutes ,which is slow going and can be very error prone.

* It is not in clinical use because it needs a large machine and it has no access to patients' record, hence, patient data has to be entered during consultation [STE92].

2.7.2 Puff Expert System[web5]

Puff (for Pulmonary Fuunction) was developed originally at Stanford University by AI experts , in conjunction with a pulmonary physician (Robert Fallate) from Pacific Medical Center whose knowledge is encoded in the program.

Puff's emphasise was on representing symbolic knowledge and using symbolic processing techniques to make decisions. Puff's medical task is to interpret respiratory at Pacific MedicalCenter. Its interpretation is based on historic and symptomatic information as wel as test results and general medical knowledge as output.

Puff typically produces a natural language explanation of its interpretation , proposes a diagnoses, and includes a confidence factor that indicates the system's confidence in that diagnose.

2.7.3 Prospector Expert System

The classic expert system that uses probabilistic reasoning is Prospector, which is designed to aid exploration geologists in determining whether a site is favorable for ore deposits or certain types. The basic idea of Prospector is to encode the expert economic geologists’ knowledge of different ore models in the system. Geologic model is a group of evidence and hypotheses that support a certain type of mineral’s being present at a site. Besides aiding in identifying minerals, Prospector could also recommend the best location to drill on the site. As more models are created, the capabilities of prospector increase [JOS98].

The program combine rule-based knowledge representation with a semantic net using certainty factors and the propagation of probabilities associated with the data [THA94].

It also use both forward and backward chaining. Forward chaining is used to select the most promising goal according to the evidence given when the system asks for them. Backward chaining used to establish or reject the goal [STE92].

CHAPTER THREE

SYSTEM DEVELOPMENT

3.1 Introduction

As mentioned in chapter one,the aim of the project is an attempt to develop an expert system to diagnose the hypomelanosis skin diseases with their treatments, which is based on an uncertainty.

The system provides a good interface for the users and human experts that enable the human experts to add, delete, update, view knowledge of a specific disease.

Two phases are considered in this system : phase one, which consist of expert interface (which used to simplify the insertion of the acquired knowledge from human experts), where it's output (knowledge base) is feed into phase two. Phase two is the executable phase at which the inference engine executes in order to get the conclusions. The inference engine also provides a method to deal with uncertainty and mechanism to handle explanations (Why and How).

The system implemented by using the visual prolog language facility which supports many advanced programming tasks such as searching, matching, and database programming, where facts are stored and can be retrieved easily.

3.2 System Architecture

The main components of the system are show in figure (3.1) .

3.2.1 User Interface

When considering the nature of the problem to be solved by an expert system, one should also consider how the end user would need to interact with the resulting system. this subsystem perform the task of communicating with the end user , at which the user may want to process an application-domain problem in a form of question – answer dialog , explanation for surprising question that the system asks, explanation of unexpected results.

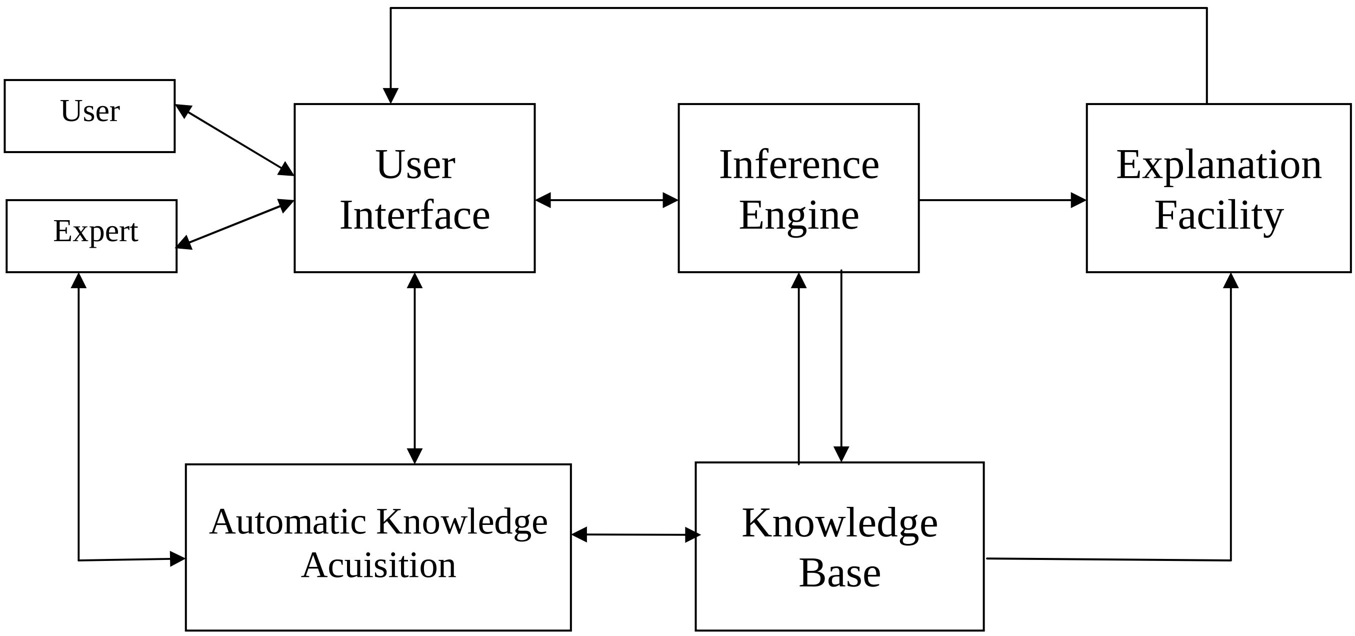


Figure (3.1) The main components of the system

The user interface is designed depending on dialog system, in which the user takes in an interactive conversation with the computer. A dialog system can be:-

*** Question- Answer Subsystem**

It used during the questioning about the information provided by the system by "yes" or "no" with certainty factor, or the questions asked to the user to identify the class of disease and so on.

*** Text system**

It is used during updating or creating new knowledge to the system.

*** Window**

The system implies of many operations like editing, creating new knowledge, help facility, explanation and so on, each can be done in separate window with title represents the specified operation.

3.2.2 Knowledge Base (KB)

It contains a description about how to solve certain problems and important facts that should be considered.

Many benefits involved in separating the knowledge base from the control [LUG89]:

- The knowledge base becomes highly modular and modifiable
- It allows the knowledge to be represented in a concise and intuitively appealing form.

It provides a great freedom in the design of knowledge representation languages.

The purpose of the K.B is to store formulated human expert knowledge about particular disease. KB. is designed to be independent from components of other subsystems. It is used by inference engine subsystem for reaching the diagnostic result after the system has inquired the answer of the relevant question.

This design depends on studying the knowledge engineering and the knowledge representation methods which they describe below:

a- Knowledge Engineering

It used for getting and formalizing the knowledge collected from the expert into a fit form in order to be expressed in an AI language.

The person who collects the knowledge called the knowledge engineer. His or her main task is to select the software and the hardware tools for the project, help the domain expert articulate the necessary knowledge and implement that knowledge in a correct and efficient knowledge base.

b- Knowledge Base Representation

The system store the human expert knowledge in an external database with it's associated B+ tree.

It uses the rule base representation to formulate the knowledge in the form of **"If….Then"** rules and store them as facts in the databases. It uses two types of databases that are provided by prolog. these are types are internal database (working memory) and external database (to store knowledge base).

The system use two internal databases, these are:

- **Evidence database:** This database is used to store the user answers about a given questions and uses this answer to check if the question is already exist, it is declared as follows:

  Database

  Evidence (question, user_answer)

- **Ques_mem:** This database is used to store the user answers about a given questions and uses this information in case of the given question is asked later, it is declared as follow:

Database

Ques_mem (question, user_answer)

The system use one external database with its associated B+tree to store the information obtained from the expert during the expert interface module.

Now see a representation of one generalized hypomelanosis skin diseases called "homosystinuria" which it may occurs in both "male" and "female" in the baby and its symptoms are:

- blond hair
- blue eyes
- CNS abnormality
- anemia
- high homocystine level

These information of the disease stored in external database as follow:

inser-part (Dis_name , Dis_gender , Dis_age , Dis_type, Terminal_list, Info_list, Treatments, Bt-sel)

Where:

Dis_name = " homosystinuria"

Dis_gender = both ("male" or "female")

Dis_age = baby

Dis_type = generalized

Terminal_list = list of symptoms of the skin disease as in homosystinuria as shown above.

Info_Ilist = this represent list of information associated to that disease, these information stored as shown in figure (3.2).

Treatment ="methionine restruction", " dietary systeine supplementation", "pyridonine suplimentation with systalhioine synthetase deficiency to reversible of hypopigmentation of hair"

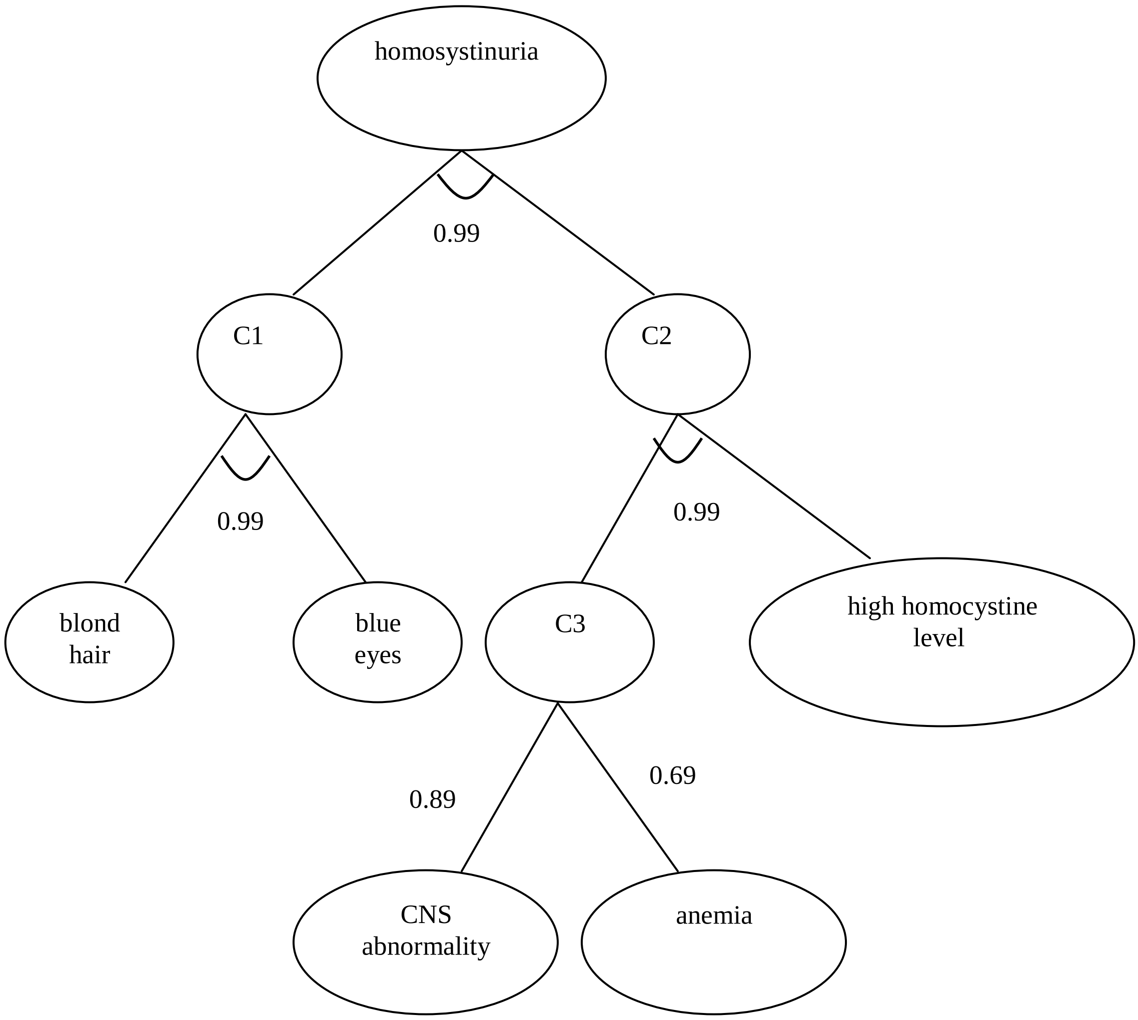


Figure (3.2) the inference network for homosystinuria disease

As an example the information stored in the list as follow:

[inf (a , homosystinuria ,pos,c1,pos,c2,0.99)

inf (a, c1,pos,blond hair,pos,blue eyes,0.99)

inf (a, c2,pos,c3,pos, high homocystine level,0.99) inf (s ,c3,pos,CNS abnormality,null,null,0.89)

inf (s ,c3,pos, anemia,null,null,0.69)]

The compound object is represented as follow:

Inf (A,B,C,D,E,F,G)

Where:

- **A** represent the relation types, if it **(a)** it mean (**And**) relation, **(o)** means (**Or**) relation and (**s**) means (**simple**) relation.
- **B** represent the node that is being reasoned about, which it is either hypothesis node or sub concluded node.
- **C**, **E** before each node either the word **(neg**) or the word (**pos)**. This is used to indicate the pattern of negation in the rule if there is any.
- **D**, **F** represent the two-piece of evidence that participate in the **(And**) or **(Or)** operation.
- **G** represents the certainty of that rule.

3.2.3 Inference Engine

The inference subsystem separated from knowledge base can enhance the power of the system, therefore the same inference engine can use but with different knowledge base.

The designed inference engine searches the external database using backward chaining technique to get the specified question and tries to prove / disprove it in order to get conclusion(s).

The inference engine in this system includes two modules:

a- Classification Module

b- Diagnostic Module

a- Classification Module

This module identifies the class index of the examined disease through three general questions before starting the diagnosis. The questions asked by the systems are: -

1-The disease type (localized, Generalized).

2-The patient age (baby, child, young, old, all).

3- The patient gender (male, female, both).

b- Diagnostic Module

Expert systems are helpful in solving diagnose problems, where the system's expertise can chose among a number of alternative decisions. Also, he can reason with uncertainty. This module include two sub modules as shown in figure (3.3)

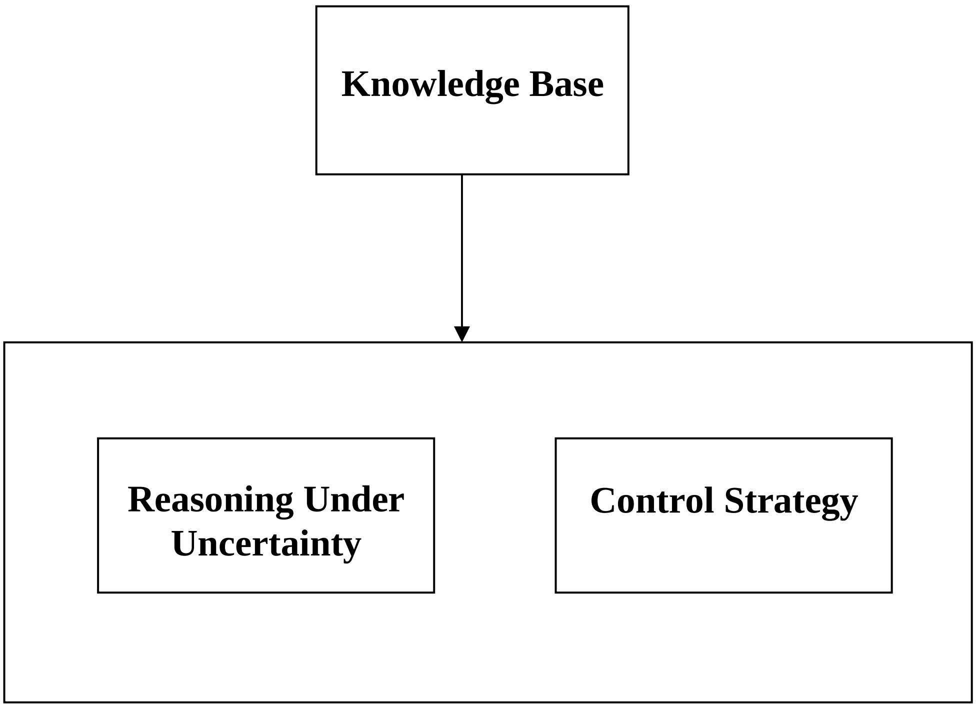


Figure (3.3) The diagnose module

- **Reasoning Under Uncertainty**

Reasoning under uncertainty is integral to all kind of diagnosing , especially the medical diagnoses, perhaps the reason is that if the physician holds off treatment until the symptoms become absolutely clear and all test results are obtained, it may then be too late to treat the patient

In this system the reasoning is based on certainty factor method to reflect uncertainty. Certainty Factor (CF) can be applied in two modes. First, it can be applied to the information supplied by the user during querying the system,

Second, it can be applied with the knowledge item, which is supplied by the expert; the level of belief supplied to the user is represented as follow:

Certainly, Too much, Much, Moderate, Little, Very little

Each one of these words has a value presented in the program as fact represented as follow:

formalize_1 (Keyword, Value)

Where the corresponding Keyword and Values are illustrated in table (3.1):

Table (3.1) The two arguments of formalize_1 predicate

Keyword	Value
certainly	0.99
too much	0.89
much	0.79
moderate	0.69

little	0.59
very little	0.49

Since the nodes (conclusion, subconclusion, premises) for each disease are represented in tree structure, each two node are connected by an Or or And operator, the CF for this rule is computed as follow:

CF (N1 And N2) = min (CF (N1), CF (N2))………….. (3.1)

CF (N1 Or N2) = max (CF (N1), CF (N2))……….….. (3.2)

Where:

N1: is the first node

N2: is the second node

To get the CF for rule's conclusion, the combined CF of the nodes (using the above combining rules) is then multiplied by the CF of the rule.

When two or more rule support the same conclusion , the certainty factor for this conclusion is calculated by using the following equations:

CF (R1) +CF (R2) - (CF (R1) × CF (R2)) when CF (R1) and CF (R2) are positive………………………………..3.3

CF (R1) +CF (R2) + (CF (R1) × CF (R2)) when CF (R1) and CF (R2) are negative…………………………….....3.4

CF (R1) + CF (R2) / 1- min (|CF (R1)| , |CF (R2)|) when either CF(R1 or CF(R2) is positive and the other is negative.……3.5

|X| is the absolute value.

- **Control Strategy**

The inference engine of the system uses the **backward chaining** as control strategy. This type of strategy is similar to the way of thinking that medical doctors used to reach its diagnosis. This strategy selects a goal and scans the rules to find those whose action part can achieve the goal. Each such rule is tested in turn. If the conditions part (collecting symptoms) for a rule matches existing facts in the working memory, the rules are applied and then diagnose the disease.

The Depth First Search (DFS) is used as a search strategy for manipulating the rules in the system. This type of search is used because it proves to be a cost effective method and can improve the search by reducing the number of rules to be searched.

In the system, the inference engine asks the user about the answer of a specific question with its certainty. This is what it doses, but first it should check its working memory to see if the answer might already be known from previous questioning. If the user supplies the information, that information will be associated in working memory so it will not have to be asked about again.

The rules used to achieve this method are: -

*** allinfer rule:** which exist as a part of all the simple **infer** rules, the allinfer rule exist to handle the situation where several separate inference rule might be applicable at particular node, invoking the (all infer) rule at a particular node cause the system to do all the reasoning possible to find out the certainty at that point.

*** infer rule**: It is used to get the certainty at a particular node and also should be able to use it at terminal node.

<u>* Combine rule:</u> which take a list of certainties and calculates a single combined certainty factor.

Each disease represented as a binary tree as shown in figure (3.2), the final goal node (homosystinuria) is the root , and the inference engine picks the first adjacent vertex which is the most left node (c1) and moves through this branch until reaches the first terminal node (blond hair), then the system will ask about this premises (blond hair ?) ,the user can answer this question by (Yes,No,Not Exact, Idont know) .

If the answer is " Yes " ,the system store certainty factor for this premises which equal to 1 .

If the answer is " No ", the certainty factor for the premises equal to -1.

If the answer is "I don't Know" ,the certainty factor equal to 0.

If the answer is "Not Exact", the user can answer this question using a spesific word such as (certainly, too much ,much,moderate, little, very little), the keyword will be converted to a real number using the predicate **formalize_1** as shown in table (3.1).

In order to establish the correctness of the sub hypothesis (c1) , the system required another piece of information to be collected , thus the system generate a question (blue eyes ?).

Since these two terminal nodes (blond hair, blue eyes) are connected by And operator , the followig rule is invoked by the inference engine in order to compute the certainty factor of this type of rule:

```
infer (Node,Terminal_list,Information_list,Ct):-

member1(inf("a",Node,S1,Node1,S2,Node2,C),Information_list),!,
        allinfer(Node1, Terminal_Node, Information_list,C1),
```

```
allinfer(Node2, Terminal_Node, Information_list,C2),
find_multiplier(S1,M1,S2,M2),
CC1=C1*M1,
CC2=C2*M2,
min(CC1,CC2,Min),
Ct=C*Min.
```

The (all infer) rule at a particular node cause the system to do all the reasoning possible to find out the certainty at that point.

The find multiplier predicates multiply the node by 1 if the sign of that node is positive, and by –1 if the sign is negative.

After the system has computed the level of confidence for the rule, it uses the DFS strategy to pick another terminal node (eg. CNS abnormality) by asking the user another question. The search continue in this mannar until all the nodes in the tree are visited and the level of believe at each node are calculated.

After all questions have been answered, the system start to calculate the final conclusion for this session (i.e the value of the root), which represent the diagnose.

The final result, which is a real-number , is converted to is equivalent keyword. This conversion is calculated in predicates called **formalize_2,** as shown below:

```
formalize_2 ("certainly",Ct):-Ct>=0.9, Ct<=1.0
formalize_2 ("too much",Ct):-Ct>=0.8, Ct<0.9
formalize_2 ("much",Ct):-Ct>=0.7, Ct<0.8
formalize_2 ("moderate",Ct):-Ct>=0.6, Ct<0.7
```

formalize_2 ("little",Ct):-Ct<=0.5, Ct<0.6

formalize_2 ("very little",Ct):-Ct>=0.4, Ct<0.5

3.2.4 Explanation Facility

The ability to justify and explain expert systems actions is usually considered as an important part of any expert system. An explanation facility is useful in many levels [PAR88]:

- It satisfies a social need by helping the end-user to feel more assured about the actions of the expert system.
- It serves as a technical tool for helping the developer to follow the operation of the expert system.

The inference system must confidence its behavior on demand to its users. This system offer two form of explanations, these forms are "HOW" and "WHY". Users may ask help from the system when they doesn't understand how the system arrived at its conclusion or why a given question is asked.

a- WHY explanation

The system needs to answer WHY? , When the system being to use an inference rule and the supporting evidence is being gathered, the rule itself must be placed on a special stack used for explanation, The Why question can be asked at any node in the KB and the system can explain why it needs the information at this node. When WHY question is asked the system pop the last inference rule from the stack and display it.

For example, if the user responds by WHY question to blond hair question that see in figure (3.2).

System: blond hair

User: why

System: I am currently trying to use a rule of the: And type to conclude (C1) which come from:

premise1: blond hair

This premise was not negated.

premise2: blue eyes

This premise was not negated.

With level of believe: certainly.

b- HOW explanation

The user can ask HOW question if he need at a particular final conclusion (disease name) where all symptoms of that disease are displayed.

3.2.5 Automatic Knowledge Acquisition

The purpose of the Automatic Knowledge Acquisition subsystem is to is to acquire knowledge from human expert in an automatic way through constructing the expert system.

In the system ,the automatic knowledge acquisition based on dialog approach, that enable the human expert to construct the knowledge base.

The automatic knowledge acquisition subsystem consist of the following sub-modules, as illustrated in figure (3.4).

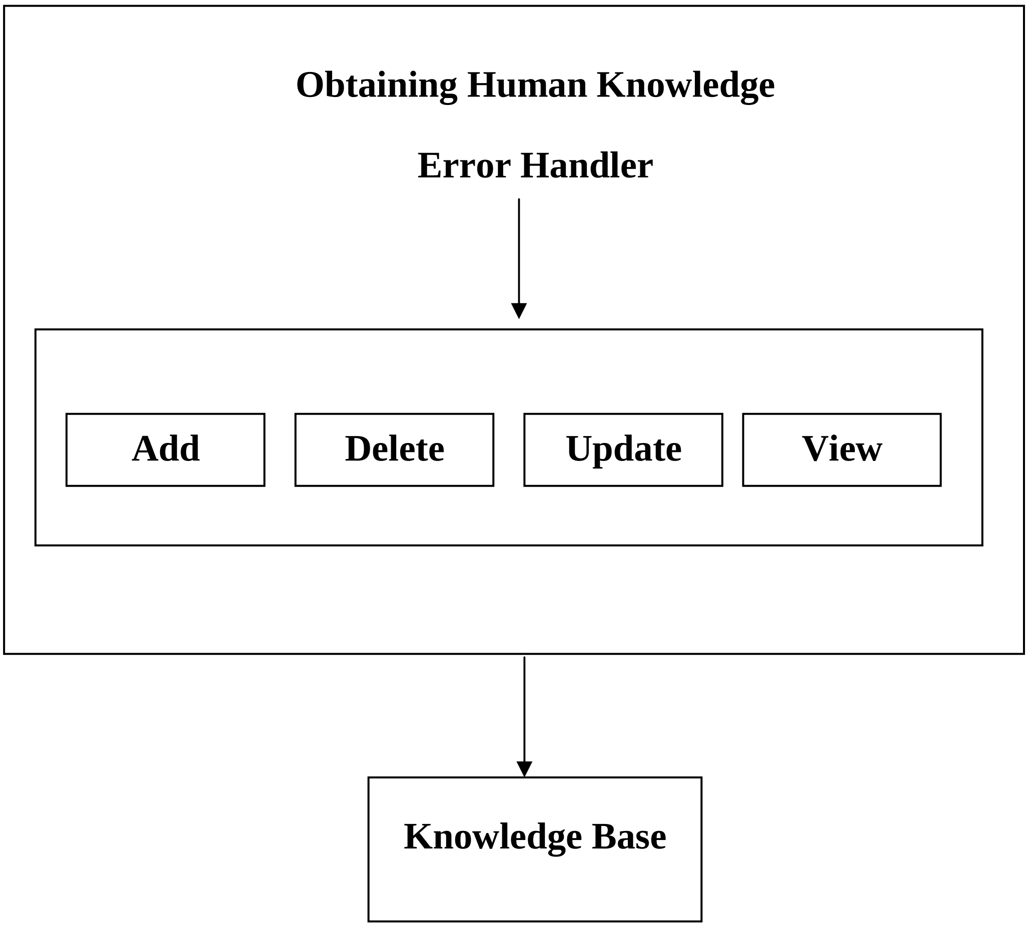


Figure (3.4) The Automatic Knowledge Acquisition

<u>1- Obtaining Human Knowledge Sub-Module</u>

It developed to allow the human expert to input the knowledge that associated to each disease as follow:

- Enter disease name
- Enter treatments
- Define number of sub concluded nodes
- Enter string that define one sub concluded node

- Define number of premises
- Enter string that define one premise
- Define number of rules
- Define rule name (Simple "s" or Or "o" or And "a")
- What is concluded from this implication?

The human expert chooses either the hypotheses node or sub concluded node.

- What the first premises condition?

The human expert chooses one premise from the list, this premise consider one terminal node to the above selected conclusion.

- Should the premises precede by not ?

If yes the human expert chose negative "neg" otherwise choose positive "pos".

- What the second premises condition?

The human expert chooses one premise from the list, this premise consider the second terminal node to the above selected conclusion.

- Should the premises precede by not ?

If yes the human expert chose negative "neg" otherwise choose positive "pos" for the second premises that related to the above implication.

- Choose the level of believe

The human expert choose the level of believe by selecting on of the words in the list, each word has a specific value stored in the program as shown in table (3.1).

2- Error Handler Sub-Module

This subsystem is designed to handle syntactic errors that may occur during the interaction between human expert and the automatic knowledge acquisition subsystem. the error handler check the following:

- The completion of knowledge: if the knowledge is incomplete, the error handler display message to human expert telling what rule is incomplete.
- if the human expert input a string value in "number of sub concluded nodes" or "number of premises " part , the error handler display message telling that invalid number.

3- Add Sub-Module

After obtaining the human knowledge, error handling operations are applied, add sub-module is called to store the human knowledge.

The human knowledge stored in external data base using the following format to create an external database:

db_create (database_name , file_name , database_location).

And create B^+-tree to store the disease name in the following format:

bt_create (database_name,
B^+-tree_name,
B^+-tree_selector,
key _length,
node_length).

The add rule sub-module using insert_part predicate to store the knowledge as follow:

```
insert_part (A,B,C,D,E,F,G,Bt_sel):-
        chain_insertz(parts,"part_chain",
        component,p (A,B,C,D,E,F,G),Ref),
        key_insert (parts,Bt_sel,A,Ref),!.
```

Where:

A: Disease name.

B: Disease gender.

C: Diseade age.

D: Disease type.

E: Symptoms list.

F: Information list: this list described in more detail in **knowledge base representation** section.

G: Treatments list.

4- Delete Sub-Module

This sub-module enables the human expert to delete any selected disease from the knowledge base.

5- Update Sub Module

This sub-module enables the human expert to update any selected disease. This can be done by selecting the item to be updated then display a message to an expert allows him to input the new symptom instead of the old one.

6- View Sub Module

This sub-module enable human expert to display the information related to the any selected disease with its picture (if available).

3.3 System Requirements

The hardware and software requirements needed to establish this proposed system are:

1- Hardware Requirements

The minimum requirements, which were used to run the system, are:

- IBM PC System or Any other compatible system (Pentiums).

- 5 MB free space in hard disk.

2- Software Requirements

-Microsoft Windows Operating system (98, Melinume, 2000, XP).

- Development environment Visual Prolog Version 5.0 or later

3.5 B^+- tree [RIC88]

In order to quick up information stored in external chains, an indexing method called B^+-tree are supports by Turbo Prolog which allows Prolog to quickly come up with a database reference number based on key that are supplied. After a database reference number has been obtained, accessing the correct term in database chain is direct and easy.

A B^+- tree is a data structure that is broken up into separate node. Each node to the tree has two nodes lower than itself, unless the node is the last one is succession. A node at the end of the tree termed a terminal node.

Terminal nodes of the tree only hold the records in B^+ -tree , these nodes are linked together to facilitate sequential processing of the records and are termed the sequential set, while non-terminal nodes are indexed to lower level , these nodes contains only key values and tree pointers.

The predicates used to manipulate the B^+-tree with their arguments are as follow:

bt_creat (DBName , Btree , BtSel , KeyLen , Order)

bt_close (DBName , BtSel)

bt_delete (DBName , Btree)

bt_open (DBName , Btree , BtSel)

key_delete (DBName , BtSel , Key , Ref)

key_first (DBName , BtSel , First)

key_insert (DBName , BtSel , Key , Ref)

key_current (DBName , BtSel , Key , Ref)

key_last (DBName , BtSel , Last)

key_next (DBName , BtSel , Next)

key_search (DBName , BtSel , Key , Ref)

Where :

DbName is the database name

Btree is a output argument, and belongs to the special domain bt_selector.When bt_creat()

is called, the third argument returns the B+_tree_selector, which is used to identify the B^+-tree.

KeyLen This arguments describes key_length. Keys are string arguments and must be large enough to allow database items to be uniquely identified. On the other hand, you won't want a key length to take up too much memory when used in conjunction with large databases.

NodeLen It determines the number of keys that that are stored on each node of the tree.

Key This argument is the key to manipulate.

Ref This argument is the associated reference number

CHAPTER FOUR
SYSTEM OPERATIONS

4.1 The operation of the system

When running the system, a main window will appear with it's bottoms, which represent the main operation of the system as follow :

1- Diagnose
2- Knowledge Base
3- Help
4- About
5- Exit

4.2.1.Diagnose

When this command is selected a new window will be activated and ask about the gender, age of the patient, and type of the hypomelanosis skin diseases. another command appears in the middle of the window which refer to start the diagnose operation as shown in figure (4.1).

When starting diagnose command initiate a sequence of questions (symptoms) of a particular disease are asked, if these symptoms are true then the disease that related with symptoms is display.

Every question is determined by several choice"Yes","No","Not exact", "Idont know","Why",

the certainty factor for these choices are as follows:

Yes	1
No	-1
I don’t know	0

In "Not Exact" the user must choose the degree of believe from the list associated with that choice, the list contain values ("certainly"," very much", "much"," moderate", "little", "very little". Figure (4.2) show the diagnose operation.

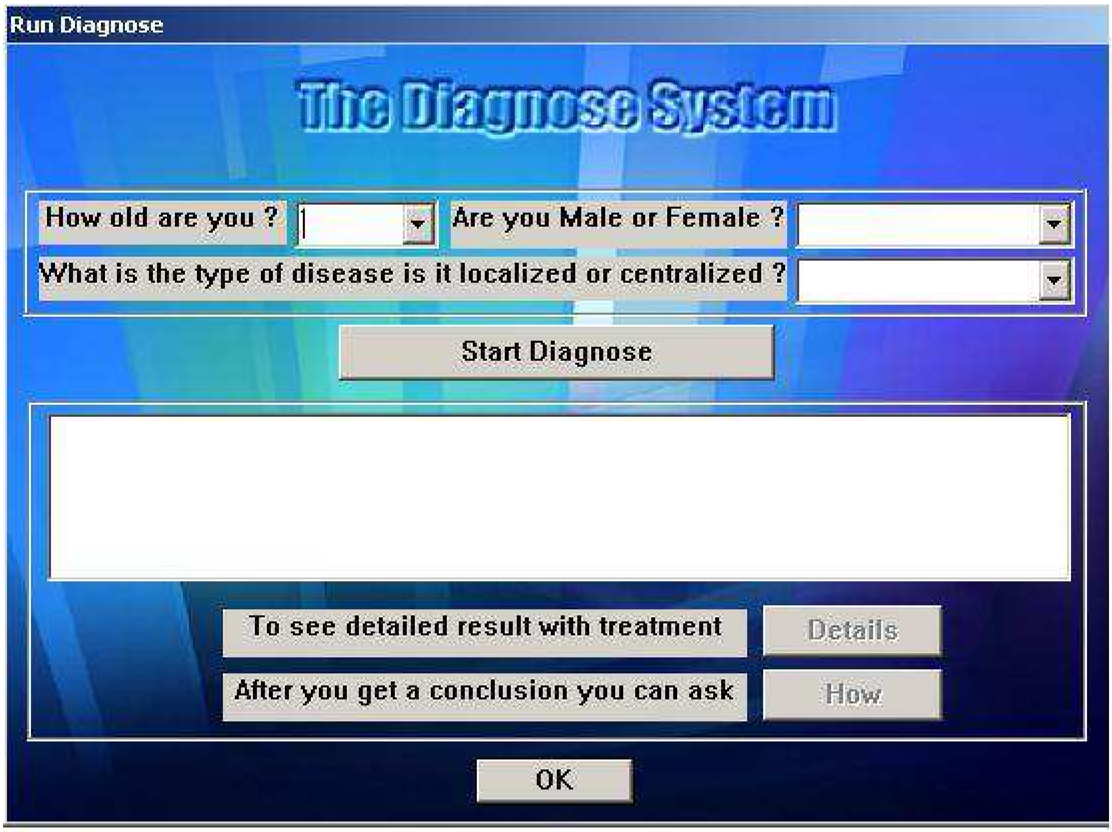


Figure (4.1) Run diagnose

Figure (4.2) Diagnose operation

After the diagnose operation is finished , the user can see the resulted disease and the possible treatments of that disease by selecting the “details” command and can ask "how" this conclusion was reached by pressing "How" command, the diseases with their symptoms and treatments see in appendix A .

4.2.2. Knowledge Base

When this command is selected, a password window will appear and a key password is asked to ensure that only the authorized person can use this command, to prevent the disease conditions being changed or deleted by any intruder. After that, this window is removed and a new one is activated which contain the essential functions of the system which include:

- Add New Knowledge.

- Update Knowledge.
- Delete Knowledge.
- View Knowledge.

1- Add New Knowledge

When this option is selected, the system ask about the disease name, gender (male, female, both), age (baby, child, old, young, all) for the patient and the type (generalized, localized) of disease, after applying these steps another window will activated to enter all the information

associated with that disease which include treatments, terminal node (symptoms) of the disease, sub concluded node, relation between two node (And, Or), if the node (Positive or Negative) and the level of believe for each sub tree.

All these information stored in external data base and the disease name stored also in B+ tree chain to specify a key which is used for quick retrieval. The add operation is shown in figure (4.3).

2- Update Knowledge

When this option is chosen the authorized person select the disease name which he (she) want to update and then the system will show all the disease symptoms with the ability to change any symptom in it.

3- Delete Knowledge

When this command is selected, the authorized person selects one of the disease names that appear in the list box to be deleted from the external database file.

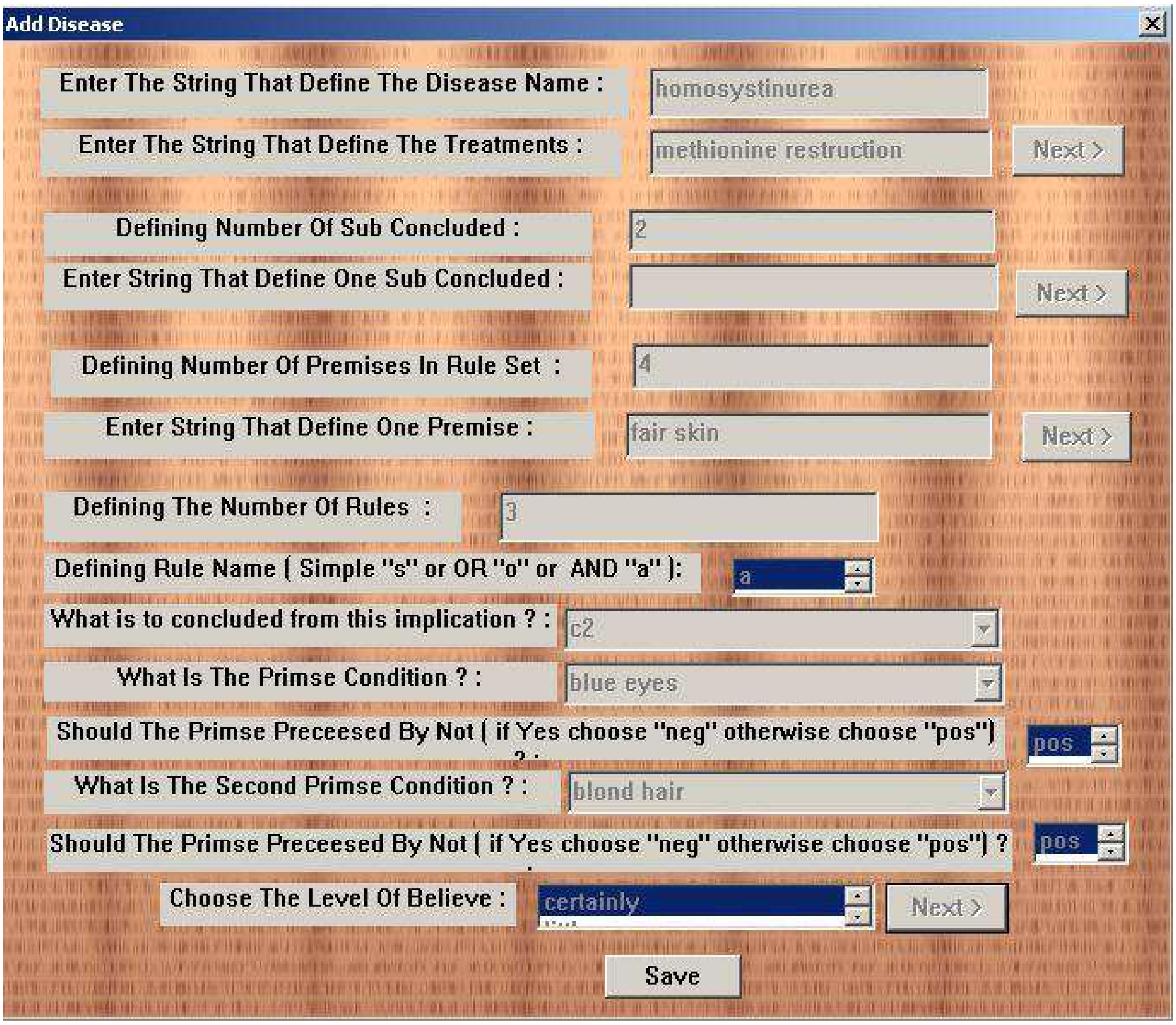


Figure (4.3) Add New Knowledge

4- View Knowledge

When this command is selected, a new window will appear, showing at the top the diseases names, the authorized person select one of these to be displayed, since the system is based on B+ tree to deal with external data base, the data retrieval will be fast, and the information of the disease will be displayed in this window as shown in the figure (4.4).

Show Picture command display the pictures associated with that disease if it exists, if there's no picture the system display the message "no picture for that disease", the pictures of some diseases appear in

appendix B

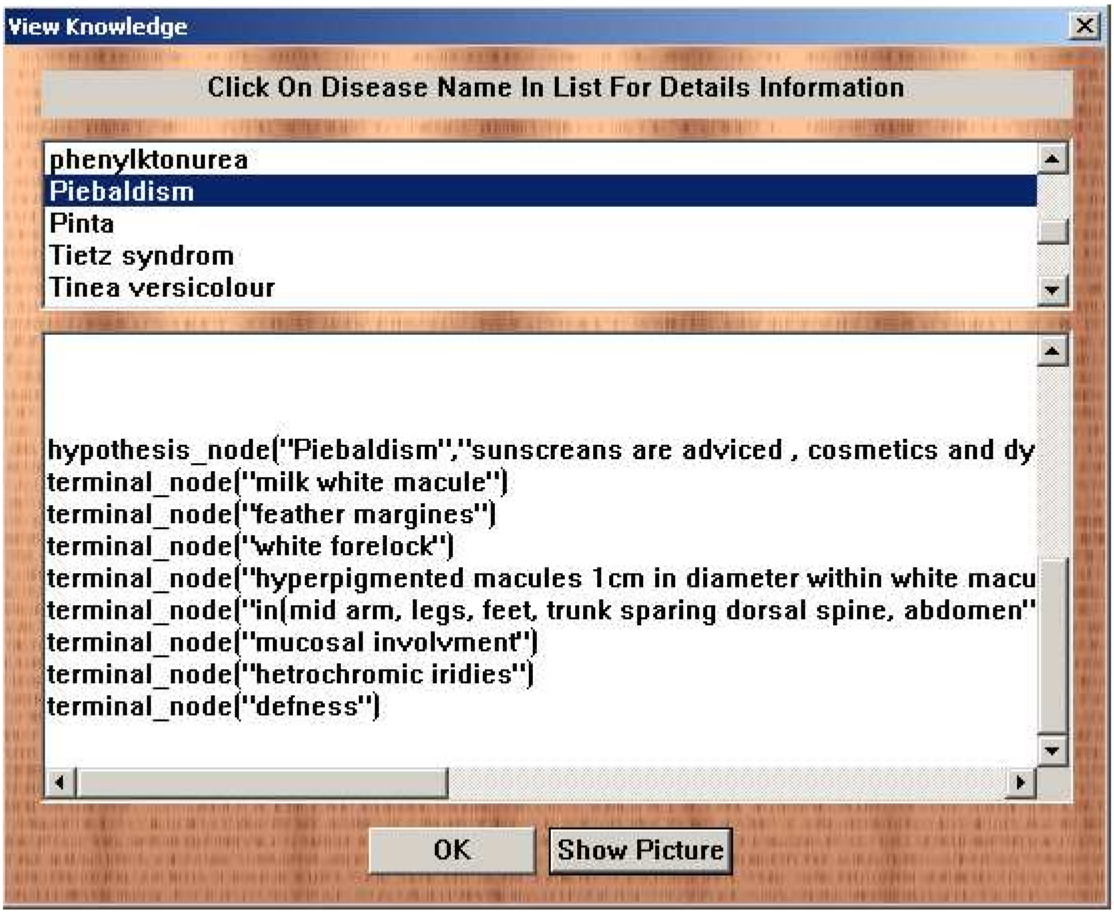


Figure (4.4) View knowledge

5- Exit

When selecting this command the system will return to the main window.

4.2.3 Help

When selecting this command, the system will display the help file (text file), which contains the features and the requirements of the system and how it run with example of how add new knowledge (Disease) to the Knowledge Base.

The work is done by using **file-str** predicate in the form

file_str ("c:\\es\\help.txt",S)

4.2.4 About

This command display general information as shown in figure (4.5).

Figure (4.5) About option

Certification of the Examination Committee

We chairman and members of the examination committee, certify that we have studied the thesis entitled **(Development Of An Expert System For The Diagnoses Of Skin Diseases)** presented by the student **Boraq Fadel Neema Al- Shannoon** and examined him its content and in what is related to it, and we have found it worthy to be accepted for the degree of Master of Science in Computer Science.

Signature:
Name: Dr. Imad Hussain Marza
Title: Assistant Professor
Date: / /2004
(Chairman)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature: Dr.Ban Nadeem Thannoon
Name:
Title: **Lecturer**
Date: **/ /2004**
(Supervisor)

Signature:
Name: **Laith Al Anne**
Title: **Dean of Collage of Science**
Date: **/ /2004**

Dedication

To my beloved

family

Acknowledgments

I would like to express my sincere gratitude and appreciation to my supervisors **Dr. Ban Nadeem Thannoon** and **Dr. Rabab Saady** for their valuable guidance, supervision and their efforts during the development of this study.

Grateful thanks for the Department of Computer Science’s stuff and employees for their support during study.

Special thanks to my friends and family for their continuous support and encouragement during the period of my study.

List of Symbol

or ~ ¬	Not
^	And
v	Or
→	Implication
>	Greater than
<	Less than
{ }	Set consisting of
|	given
∑	Sum of
θ	Frame of discernment
⊆	Subset of
∈	

References

[ALK02] **AL-Kateeb B.I.,** "Decision and Implementation of Diagnosing Expert System Shell", MSC thesis, Saddam University, Department of computer science, 2002.

[BRA01] **Bratko,I.,** "Prolog programming for Artificial intelligence", 3rd Edition, Pearson Education Limited, 2001.

[DAR00] **Darlington K.,** "The essence of Expert systems", Prentice Hall, 2000.

[FOR89] **Forsyth,R.,** "Expert System:Principles and case studies", Chapman and Hall Publishing Company,USA,2nd Edition,1989.

[Ger01] **Gerges G**., **"**An Expert System Solution to Anaerobic Bacterial Identifications", MSC thesis, Houston University, department of Computer Science, 2001(Internet).

[JOS98] **Joseph G.**, **Riley, Gary**; "Expert System Principle and Programming", 3rd Edition, PWS Publishing Company, 1998.

[KAN98] **Kandela H.F.,** "Design of Expert SystemUsing Neural Networks", PhD Thesis,Saddam University,Department of Computer Science,1998.

[LUG98] **Luger, G.F.,**"Artificial Intelligence", Addison Wesley J. ongman.Inc, 1998.

[LUG89] **Luger, G. F.,** William, "Artificial Intelligent and the design of Expert System", The Benjamin/Cummings Publishing Company,1989.

[MAR89] **Marchus D.H.,** "Expert System Programming in Turbo Prolog", Prentic Hall, 1989.

[NIE01] **Niederlinski A.,** "An Expert System Shell For Uncertainm Rule and Model-Based Reasoning", Department of Automation, Computer Science and Electronics, Silesian University of Technology, Akademicka 16, 44

[PAR88] **Parsaye,K. , Chignell,.M,** "Expert System for experts", John Wiley and Sons Inc., 1988.

[RIC91] Rich E. ,Knght.K., "Artificial Intelligence",McGraw-Hill,1991.

[RIC88] **Rich K., Robinson P.**,"Using Turbo Prolog",McGraw-Hill,1988.

[STE92] **Stephan J.J.,"**Certainty factor based expert system for diagnosing recurrent abdominal pain", MSC thesis, Baghdad University, Department of computer science,1992.

[SRI01] **Srinivas Y., D.W.T, D.John**,"Acomprative study of three expert systems for blood pressure control",2001 (Internet).

[TUR98] **Turban E., Aronson J.E.,**"Decision support system and intelligent systems", Prentic-Hall, 1998.

[THA94] **Thannoun B.N.,**"An Expert System Approach to an application in Iraq law", MSC thesis, Saddam University, department of computer science, 1994.

[TOB91] **Tobia W.F.,** "Pc-Based Expert System for Logic Design", MSC thesis, Technology University, Department of Electrical Engineering, 1991.

[TOW88] **Townsend,C.,** "introduction to turbo prolog", Business Promotion Burean Publications, Delhi,1988.

[WAT86] **Waterman D.A.,**"Aguid to Expert System",Addition Publishing Company",1986.

[WEI88] **Weiskamp,K. and Hengle,T. ,**"Artificial Intelligence Programming with Turbo Prolog", John-Wilely and Sons ,Inc.,USA,1988.

[web1] http://www.cs.msstate.edu/~hansen/classes/Alfallo2/slides/ expert system.pdf.

[web2] www.cs.bham.ac.uk/~jxb/ALTA/w10s3.pdf,2000

[web3] www.ncst.ernet.in/kbcs/virek.

[web4] www.idi.ntnu.no/~ksys/notes/cfmodel.htm

[web5] www.ee.vt.edu/~ee4524hv/slides9.pdf

Supervisor Certification

We certify that this thesis was prepared under our supervision at the Department of Computer Science / College of Science / Al-Nahrain University, by **Boraq Fadel Neema Al- Shannoon** as a partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Signature:

Name: **Ban Nadeem Thannoon**

Title: **Assist Proff.**

Date: / /2004

In view of the available recommendations, I forward this thesis for debate by the examination committee.

Signature:

Name: **Dr. Taha S. Bashaga**

Title: Head of the Department of Computer Science, Al-Nahrain University.

Date: / / 2004

Certification of the Examination Committee

We chairman and members of the examination committee, certify that we have studied the thesis entitled **(Development Of An Expert System For The Diagnoses Of Skin Diseases)** presented by the student **Boraq Fadel Neema Al- Shannoon** and examined him its content and in what is related to it, and we have found it worthy to be accepted for the degree of Master of Science in Computer Science.

Signature:
Name:
Title: Assistant Professor
Date: / /2004
(Chairman)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Member)

Signature:
Name:
Title:
Date: **/ /2004**
(Supervisor)

Signature:
Name:
Title: **Lecturer**
Date: **/ /2004**
(Supervisor)

Signature:
Name: **Laith ?. ?.**
Title: **Dean of Collage of Science**
Date: **/ /2004**

Dedication

To my beloved family

Acknowledgments

I would like to express my sincere gratitude and appreciation to my supervisors **Dr. Ban Nadeem Thannoon** and **Dr. Rabab Saady** for their valuable guidance, supervision and their efforts during the development of this study.

Grateful thanks for the Department of Computer Science's stuff and employees for their support during study.

Special thanks to my friends and family for their continuous support and encouragement during the period of my study.

Table Of Contents

Republic of Iraq
Al Nahrain University
College of Science

Development Of An Expert System For the Diagnoses of Skin Diseases with the ways of Treatments

A Thesis
Submitted to the College of Science, Al-Nahrain University in Partial Fulfillment of the Requirements for The Degree of Master of Science in Computer Science

BY
Boraq Fadel Neama Al- Shannoon
(B. Sc. 2001)

Rabia Al-Awal 1425 AH May 2004 AC