What is an Expert System?
A rule based computer system which simulates a human expert in his/her field of expertise in an attempt to solve a particular problem.

An expert system (commonly referred to as a knowledge based system):

- Accumulates the knowledge of human experts or the knowledge obtained from a variety of different “expert” sources (e.g. documentation).
- The decisions (conclusions) of expert systems are based on:
  - The accumulated knowledge (represented as facts)
  - Rules operating on the existing facts.

An example of facts and rules

**Facts:**
- Engine off.
- Pedestrian crossing the road in a distance less than 10 meters away.

**Rules:**
- if (car needs to be driven) and (engine off) then
  - start engine
end
- if (a pedestrian is crossing the road) and (in a distance less than 10 meters away) then
  - break
end

Advantages of Expert Systems over Human Experts

- An expert system can operate constantly 24 hours per day.
- An expert system can exceed the performance of any human expert, as it can combine knowledge from a number of different experts.
Characteristics of an Expert System

Expert systems should be:

- Goal oriented, i.e. their actions (questions to the user) are based on the desired goal(s) that the system must achieve.
- Able to work with imperfect information. If some specific information is missing the expert system must choose an alternative route (question to the user) to reach a decision.
- Able to cope with different combinations of events, independent of the order that these appear.
- Able to justify (give the reasons) for their decision/recommendation.

Requirements for replacing an expert with an expert system

To be able to replace an expert with a computerised expert system, the expert must:

- be able to perform the task.
- know how they perform the task.
- be able to explain how they perform the task.
- have the time to describe in detail how they perform the task.
- be willing to explain how they perform the task.

The above are included in the limitations (disadvantages) of expert systems.

A typical scenario - Diagnosis of Car Problems over the Phone

Mechanic: Hello, how may I help?

Driver: My car wouldn’t start.
Mechanic: Have you checked that you aren’t out of petrol?
Driver: I’m not certain the tank is empty.

Mechanic: As you crank the starter, do you smell petrol?
Driver: No, I turned it over for a long time, but didn’t smell anything.

Mechanic: Based on what you’ve told me, I’m almost certain your car is out of petrol.
Driver: Thanks for the advice. Mind telling me how you reached your conclusion?

Mechanic: When a car won’t start my initial suspicion is that the battery is dead, the starter has failed or some other electrical problem exists.
Your input that the starter operates makes it more likely that no fuel is getting to the engine.
Although you are not sure that the petrol tank is empty, the fact that you don’t smell petrol when the engine turns over supports my conclusion that you are out of petrol.
Applications of Expert Systems

Some of the applications of expert systems include:

- **Product manufacturing and configuration**: A product being manufactured or sold depends on a number of parts which need to be included. Each of these parts depends on other components.

- **Mail filtering**: Messages can be deleted or saved in a specific directory based on their header/subject/content.

- **Implementing business rules**: Business rules are strategies that a company applies in either its products or its operation. Computerised systems should be able to follow and check the validity of these rules. An example of this can be based on web applications. Consider the recommendations that could be made to a buyer of a specific product, based on the preferences of buyers of the same product.

Declarative programming vs Procedural/Object Oriented Programming

- In procedural and object oriented programming, a program instructs the computer:
  - what to do
  - how to do it
  - in what order

- In declarative programming, a program describes to the computer:
  - what to do, without a detailed plan of how to do it.

The Architecture of a rule based system

An expert system consists of:

- **An inference engine (rule engine)**: applies rules to data (facts).
  - pattern matcher
  - agenda
  - execution engine

- **A rule base**: contains the rules.

- **A working memory**: stores the data that the inference engine operates on.
Case Study: MYCIN

MYCIN developed in the 1970s was an expert system which is responsible for the diagnosis and recommendation of treatment for some blood infections.

- Its knowledge was represented as a set of rules, similar to:
  
  if the infection is x and
  the morphology of the organism is y and
  the aerobicity of the organism is z

  then

  there is suggestive evidence (0.8) that
  infection is of w type and
  that the class of the organism is v.

- 0.8 describes the probability that the conclusion is certain.

- MYCIN was written in Lisp, but it was never placed in practice for ethical and legal reasons.

A Puzzle that an expert system can solve

Consider the following puzzle:

Four players of golf are standing in a line from left to right. They are supposed to play in a particular order. Each golfer wears different colour pants.

1. One is wearing red.
2. The golfer next to Fred’s immediate right is wearing blue.
3. Joe is second in line.
4. Bob is wearing plaid pants.
5. Tom isn’t in position one or four, and he isn’t wearing orange.
6. In what order the four golfers will play, and what is colour of each golfer?

Clearly, there is no direct way (algorithm) so as to write a procedural or object oriented program to solve it. Expert systems can solve it, by just representing these facts in an appropriate way and express some simple rules which connect facts to newly created facts, in order to draw a conclusion.

Limitations of Expert Systems

Some of the limitations of expert systems include:

- Not able to cope with unseen information.
- Not able to learn independently in a similar manner that humans learn. They need to be programmed in advance.
- Not able to cope with noise.
- Not able to adapt to new environments.

Many of these limitations are not present in machine learning techniques, e.g. neural networks.