1. Designing a rule-based Expert System

Problem Description

```
To design a rule based Expert System capable of deduction and explaining rulesets
and designing a suitable shell to use the system.
```

Formal Problem Statement

- Purpose:

  To accept data for ultimate stress, relative density and cost data and return what material is suitable for making a beam given ultimate stress, relative density and cost data.

- Input / The data:

  The data required to create such a Expert System consists of:
  - **A rulebase** containing rules to be applied to decide on the material.
  - **Global Database** containing the facts or assertions about the particular problem being solved.
  - **Parameters** for which values are provided by the user. Used for giving results.

- Features:

  - The program features a interactive console for the user. The user is asked several questions which would be required to identify the parameters of the problem and hence correct material as the result.
  - In this buffer, the user can also ask the system "*why*" a certain value is required and for a more precise understanding, ask what "*rule*" is being currently applied.
  - If the user doesn't know the value of a certain parameter or the value of the parameter isn't relevant to the user, the program will accept "*unknown*" to understand this.
  - The program would also make certain that the values entered by the user is a valid input.

- Output

  - The programs outputs the suitable material along with its certainty factor. If multiple options exist, all of them are printed. If none of the materials satisfy, the programs returns failure.

2. AI Modelling

The program uses generalized heuristic reasoning and constraint satisfaction techniques using certainty factors of a possible outcome to check the applicability of rules and decide what material best fits the needs mentioned.

It also uses forward chaining for creating new rules upon the ones provided in the rulebase.

3. Solution Approach:

> The program is written in python3. No extra modules were used.

The basic functioning of the program was inspired by emycin and hence it also uses contexts instead of variables to represent states. This context is a class for which individual instances for different entities can be created. These instances are characterized by parameters which define a certain characteristic of the instance.

To operate on the user provided data, the global database also contains various rules consisting of one or more conditions/propositions which operates on the certainty factors.

> Operations on certainity factor such as `or` and `and` are done using simple Bayes Theorem in order to keep the program less complicated.

It is also capable of backward chaining i.e. the programs rejects a possibility whenever the certainty of it shows that it is absolutely false.

The shell object is responsible for interacting with the user. Thus the shell object is also used to hold the definitions of contexts, their parameters and the rules applicable. These are pre-defined as the rulebase and are associated with a shell object. The shell is capable of answering which rule is being considered and also explain why it is being considered in some detail pointing out what factors are known and are contributing to the certainty of applicability of the rule.

4. References:

- Principles of Rule based expert systems - Standford Univ.
- https://www.sciencedirect.com/topics/computer-science/certainty-factor
- Paradigms of AI programming - Peter Norvig