**EXPERIMENT NO : 09 DATE : 23/04/24**

**Aim**: Implement an Expert System for Health Diagnosis

**Theory**:

• An expert system is a computer program that is designed to solve complex problems and to provide decision-making ability like a human expert.

• It performs this by extracting knowledge from its knowledge base using the reasoning and inference rules according to the user queries.

• The expert system is a part of AI, and the first ES was developed in the year 1970, which was the first successful approach of artificial intelligence.

• It solves the most complex issue as an expert by extracting the knowledge stored in its knowledge base.

• The system helps in decision making for complex problems using both facts and heuristics like a human expert.

• It is called so because it contains the expert knowledge of a specific domain and can solve any complex problem of that particular domain.

• These systems are designed for a specific domain, such as medicine, science, etc.

• The performance of an expert system is based on the expert's knowledge stored in its knowledge base.

• The more knowledge stored in the KB, the more that system improves its performance.

• One of the common examples of an ES is a suggestion of spelling errors while typing in the Google search box.

Examples Of Expert Systems:

• DENDRAL: It was an artificial intelligence project that was made as a chemical analysis expert system. It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.

• MYCIN: It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis. It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases.

• PXDES: It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm.

• CaDeT: The CaDet expert system is a diagnostic support system that can detect cancer at early stages.

Building An Expert System:

• Determining the characteristics of the problem.

• Knowledge engineer and domain expert work in coherence to define the problem.

• The knowledge engineer translates the knowledge into a computer-understandable language.

• He designs an inference engine, a reasoning structure, which can use knowledge when needed.

• Knowledge Expert also determines how to integrate the use of uncertain knowledge in the reasoning process and what type of explanation would be useful.

**CODE**

def health\_diagnostic():

print("Welcome to the Health Diagnostic System!")

potential\_diseases = []

# Ask questions based on the knowledge base

print("Do you have a fever?")

fever = input("Enter 'yes' or 'no': ").lower()

if fever == "yes":

print("Do you have a cough?")

cough = input("Enter 'yes' or 'no': ").lower()

if cough == "yes":

print("Do you have shortness of breath?")

breathlessness = input("Enter 'yes' or 'no': ").lower()

if breathlessness == "yes":

potential\_diseases.append("COVID-19")

else:

print("Do you have chest pain or discomfort?")

chest\_pain = input("Enter 'yes' or 'no': ").lower()

if chest\_pain == "yes":

potential\_diseases.append("Bronchitis")

else:

print("Do you have fatigue or body aches?")

fatigue = input("Enter 'yes' or 'no': ").lower()

if fatigue == "yes":

potential\_diseases.append("COVID-19")

else:

print("Do you have headache or confusion?")

confusion = input("Enter 'yes' or 'no': ").lower()

if confusion == "yes":

potential\_diseases.append("COVID-19")

else:

print("Do you have sore throat or runny nose?")

throat\_nose\_symptoms = input(

"Enter 'yes' or 'no': "

).lower()

if throat\_nose\_symptoms == "yes":

potential\_diseases.append("COVID-19")

else:

potential\_diseases.append("COVID-19")

else:

print("Do you have muscle aches or fatigue?")

muscle\_aches = input("Enter 'yes' or 'no': ").lower()

if muscle\_aches == "yes":

potential\_diseases.append("COVID-19")

else:

print("Do you have sore throat or runny nose?")

throat\_nose\_symptoms = input("Enter 'yes' or 'no': ").lower()

if throat\_nose\_symptoms == "yes":

potential\_diseases.append("Common Cold")

else:

potential\_diseases.append("COVID-19")

else:

print("Do you have a sore throat?")

sore\_throat = input("Enter 'yes' or 'no': ").lower()

if sore\_throat == "yes":

print("Do you have swollen glands in your neck?")

swollen\_glands = input("Enter 'yes' or 'no': ").lower()

if swollen\_glands == "yes":

potential\_diseases.append("Strep Throat")

else:

print("Do you have white patches on your tonsils?")

white\_patches = input("Enter 'yes' or 'no': ").lower()

if white\_patches == "yes":

potential\_diseases.append("Strep Throat")

else:

potential\_diseases.append("Sore Throat")

else:

print("Do you have chest congestion or wheezing?")

chest\_congestion = input("Enter 'yes' or 'no': ").lower()

if chest\_congestion == "yes":

potential\_diseases.append("Bronchitis")

else:

print("Do you have body aches or joint pain?")

body\_joint\_pain = input("Enter 'yes' or 'no': ").lower()

if body\_joint\_pain == "yes":

potential\_diseases.append("COVID-19")

else:

print("Do you have difficulty swallowing or loss of appetite?")

swallowing\_difficulty = input("Enter 'yes' or 'no': ").lower()

if swallowing\_difficulty == "yes":

potential\_diseases.append("Strep Throat")

else:

potential\_diseases.append("No specific disease")

# Print potential diseases

if potential\_diseases:

print("Based on your symptoms, potential diseases may include:")

for disease in potential\_diseases:

print("- " + disease)

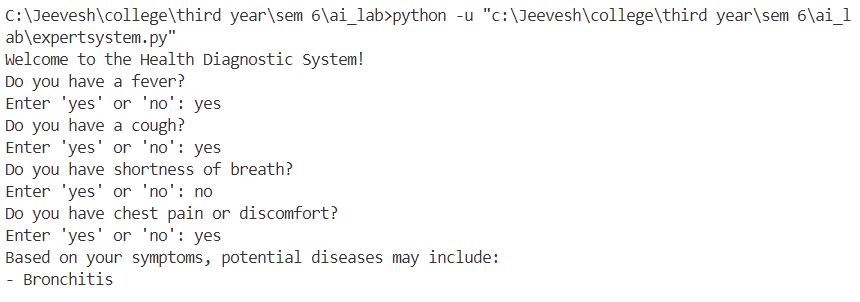
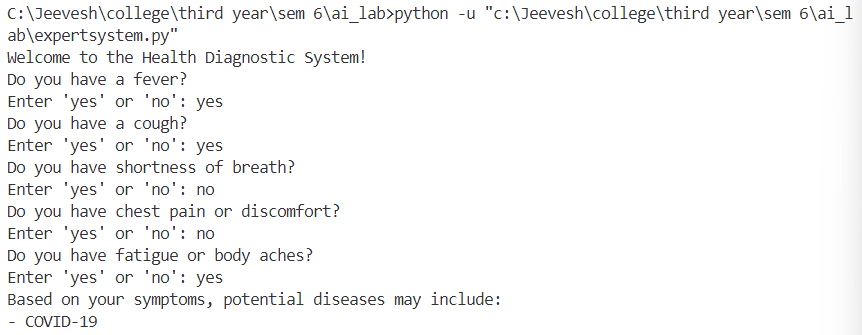
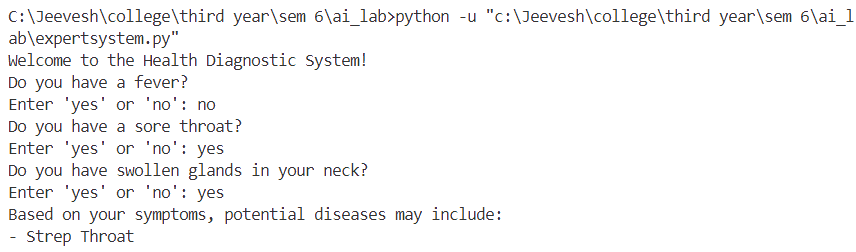
else:

print("No potential diseases found based on your symptoms.")

# Call the function to start the diagnostic process

health\_diagnostic

**Output:**

****

**CONCLUSION:**

Implementation of the Expert System Problem was carried out by tracing the algorithm and above output was obtained during the said execution.