KINGDOM OF SAUDI ARABIA
Ministry of Higher Education
Taibah University
College of Computer Science and
Engineering (Girls Section)

المملكة العربية السعودية
وزارة التعليم العالي
جامعة طيبة
كلية علوم وهندسة الحاسب الالي

Mushrooms classification system

CS 362 - INTELLIGENT SYSTEMS

Section: C8A

Norah Fahad Aloufi | 4050772

Shatha Salem Alreheli | 4051126

Rahaf Abdulrahman Allogmani | 4051106

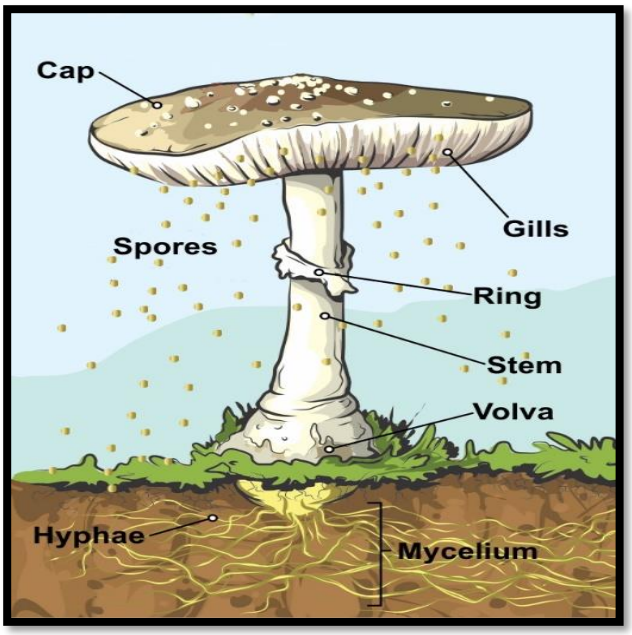
Instructor: Ghada Alharbi

Spring 2022

Introduction and Description of the problem

An expert system is AI software system that uses a content to resolve complicated issues. in an exceedingly rule-based knowledge system, the data is pictured as a collection of rules. every rule specifies a relation, recommendation, directive, strategy, or heuristic and has the IF (condition) THEN (action) structure.

This project focuses on an expert system for Mushrooms classification as poisonous or edible by employing a content (facts and rules) we tend to classify mushrooms consistent with totally different options like cap-color, gill-color, cap-shape...etc. These options can verify whether the mushroom is edible or poisonous.



Knowledge Base

Facts:

Edible mushroom has Presence Worms
Edible mushroom has Presence Scales
Edible mushroom has pleasant Odor
Edible mushroom has sweet Taste
Edible mushroom has green Cap Color
Edible mushroom has sunken Cap Shape
Edible mushroom has red or orange Gill Color
Edible mushroom has brown Veil Color
Edible mushroom has flaring Ring
Edible mushroom has black, orange, purple or yellow Spore Color

Poisonous mushroom has not Presence Worms
Poisonous mushroom has not Presence Scales
Poisonous mushroom has bad Odor
Poisonous mushroom has bitter Taste
Poisonous mushroom has bad Taste
Poisonous mushroom has sour Taste
Poisonous mushroom has purple Cap Color
Poisonous mushroom has convex Cap Shape
Poisonous mushroom has buff or green Gill Color
Poisonous mushroom has yellow Veil Color
Poisonous mushroom has not Ring

Poisonous mushroom has green Spore Color
Poisonous mushroom has grooves Cap Surface

Rules:

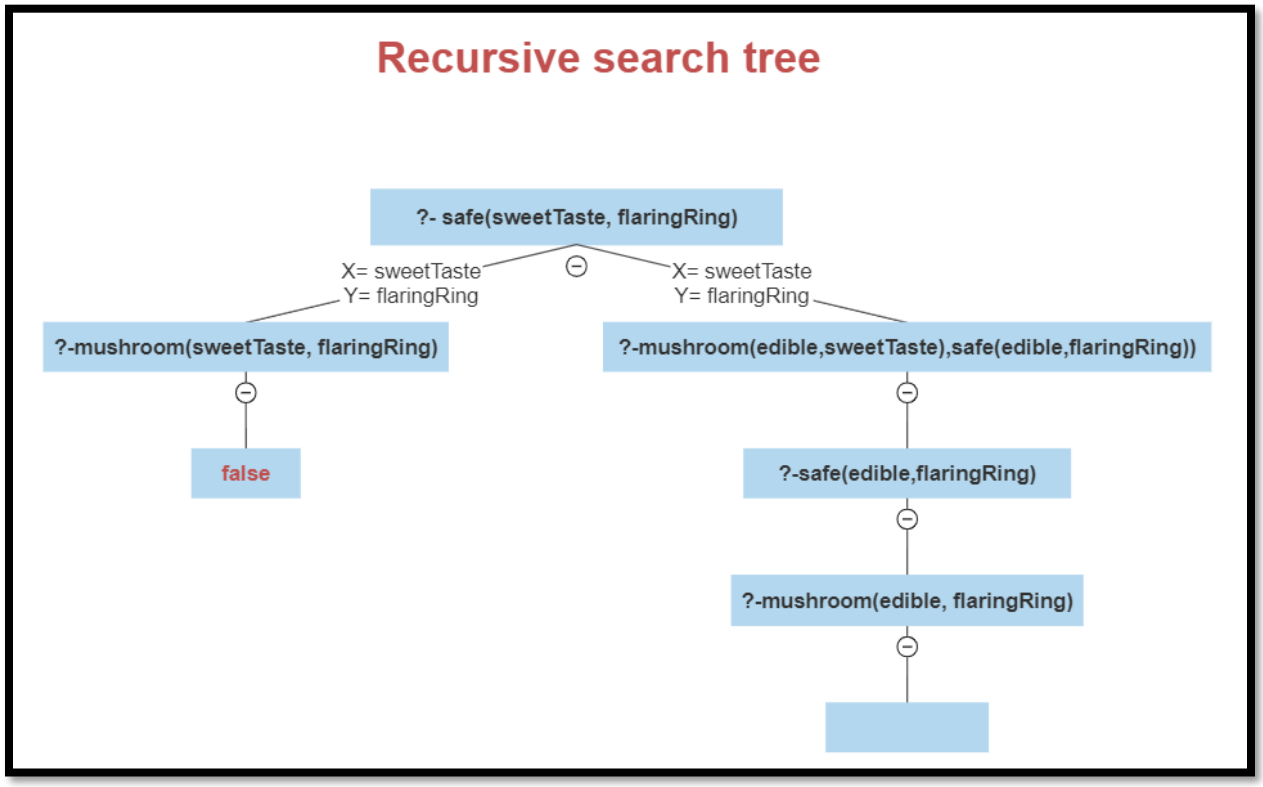
Mushroom are safe if its properties are for edible mushroom.
Mushroom are not safe if its properties are for poisonous mushroom.

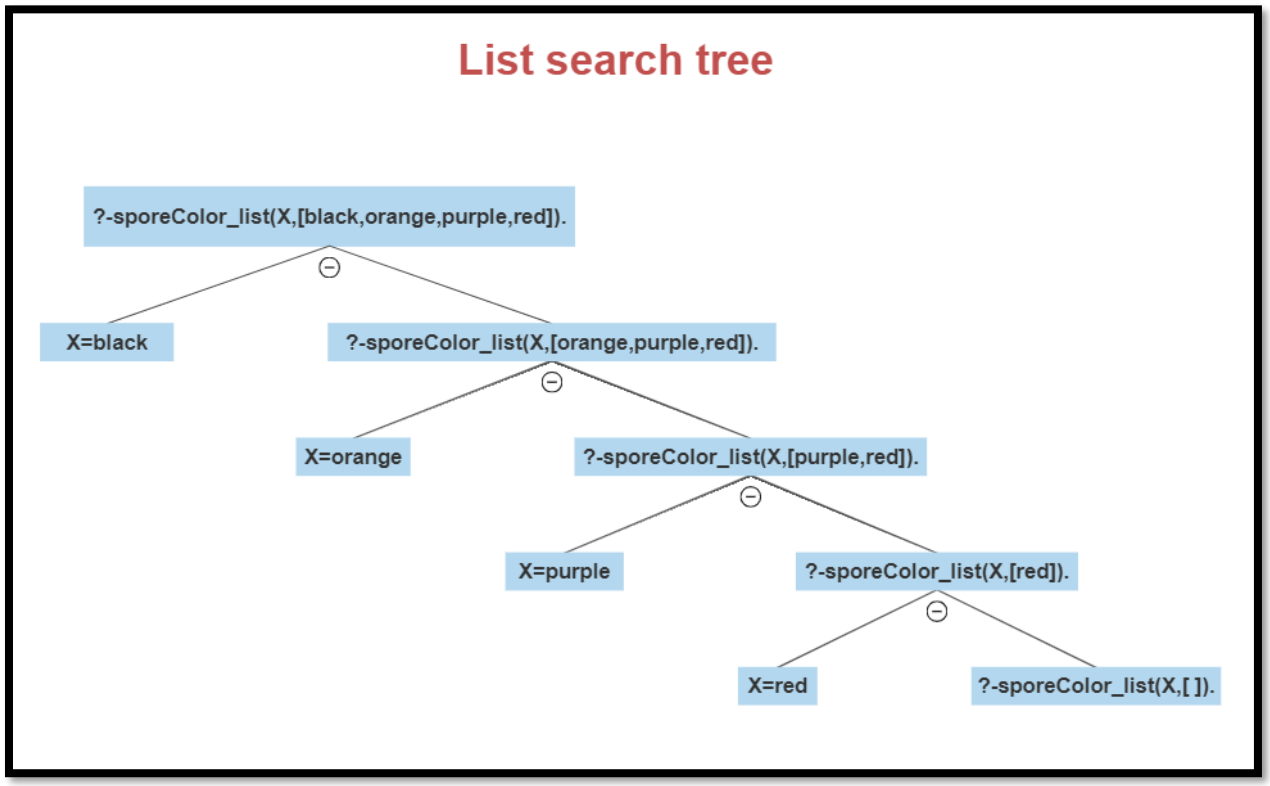
Codes

```
/*FACTS*/

mushroomType(poisonous).
mushroomType(edible).

mushroom(edible,hasPresenceWorms).
mushroom(edible,hasPresenceScales).
mushroom(edible,pleasantOdor).
mushroom(edible,sweetTaste).
mushroom(edible,greenCapColor).
mushroom(edible,purpleCapColor).
mushroom(edible,sunkenCapShape).
mushroom(edible,redGillColor).
mushroom(edible,orangeGillColor).
mushroom(edible,brownVeilColor).
mushroom(edible,flaringRing).
mushroom(edible,blackSporeColor).
mushroom(edible,orangeSporeColor).
mushroom(edible,purpleSporeColor).
mushroom(edible,yellowSporeColor).

mushroom(poisonous,noPresenceWorms).
mushroom(poisonous,noPresenceScales).
mushroom(poisonous,groovesCapSurface).
mushroom(poisonous,badOdor).
mushroom(poisonous,bitterTaste).
mushroom(poisonous,sourTaste).
mushroom(poisonous,badTaste).
mushroom(poisonous,convexCapShape).
```

```
mushroom(poisonous,buffGillColor).
mushroom(poisonous,greenGillColor).
mushroom(poisonous,yellowVeilColor).
mushroom(poisonous,noRing).
mushroom(poisonous,greenSporeColor).

/*RULES*/
edible(X):-write('Edible mushroom properties?'),mushroom(edible,X).
poisonous(X):-write('Poisonous mushroom
properties?'),mushroom(poisonous,X).

safe(X):-mushroom(edible,X).
notSafe(X):-mushroom(poisonous,X).

/*recursive*/
safe(X,Y):-mushroom(X,Y).
safe(X,Y):-(mushroom(edible,X),safe(edible,Y)).

notSafe(X,Y):-mushroom(X,Y).
notSafe(X,Y):-(mushroom(poisonous,X),notSafe(poisonous,Y)).

/*List*/
sporeColor_list(X,[X|_]).
sporeColor_list(X,[_|T]):- sporeColor_list(X,T).

listOfMushroomProperties([presenceWorms,presenceScales,odor,taste,capC
olor,capShape,gillColor,veilColor,ring,sporeColor,capSurface]).
mushroomProperties(X):-listOfMushroomProperties(T),member(X,T).
```

Search Tree


Recursive search tree
?- safe(sweetTaste, flaringRing)
X= sweetTaste
Y= flaringRing
X= sweetTaste
Y= flaringRing
?-mushroom(sweetTaste, flaringRing)
?-mushroom(edible,sweetTaste),safe(edible,flaringRing))
false
?-safe(edible,flaringRing)
?-mushroom(edible, flaringRing)


List search tree
?-sporeColor_list(X,[black,orange,purple,red]).
X=black
?-sporeColor_list(X,[orange,purple,red]).
X=orange
?-sporeColor_list(X,[purple,red]).
X=purple
?-sporeColor_list(X,[red]).
X=red
?-sporeColor_list(X,[ ]).

Sample outputs

```
?- edible(X).
Edible mushroom properties?
X = hasPresenceWorms ;
X = hasPresenceScales ;
X = pleasantOdor ;
X = sweetTaste ;
X = greenCapColor ;
X = purpleCapColor ;
X = sunkenCapShape ;
X = redGillColor ;
X = orangeGillColor ;
X = brownVeilColor ;
X = flaringRing ;
X = blackSporeColor ;
X = orangeSporeColor ;
X = purpleSporeColor ;
X = yellowSporeColor.

?- poisonous(X).
Poisonous mushroom properties?
X = noPresenceWorms ;
X = noPresenceScales ;
X = groovesCapSurface ;
X = badOdor ;
X = bitterTaste ;
X = sourTaste ;
X = badTaste ;
X = convexCapShape ;
X = buffGillColor ;
X = greenGillColor ;
X = yellowVeilColor ;
X = noRing ;
X = greenSporeColor.
```

```
?- safe(orangeGillColor,pleasantOdor).
true .

?- safe(orangeGillColor,badOdor).
false.

?- safe(yellowVeilColor,badOdor).
false.

?- notSafe(yellowVeilColor,badOdor).
true .

?- notSafe(sunkenCapShape,badOdor).
false.

?- notSafe(sunkenCapShape,sweetTaste).
false.
```

```
?- mushroomProperties(X).
X = presenceWorms ;
X = presenceScales ;
X = odor ;
X = taste ;
X = capColor ;
X = capShape ;
X = gillColor ;
X = veilColor ;
X = ring ;
X = sporeColor ;
X = capSurface.
```

```
?- safe(flaringRing).
true.

?- safe(sourTaste).
false.

?- notSafe(sourTaste).
true.

?- notSafe(hasPresenceWorms).
false.
```