UNIVERSITY OF ASIA PACIFIC

Department of Computer Science of Engineering

Lab Mid Test:

Part-1

Course code: CSE(404)

Course Title: Artificial Intelligence and Expert System Lab

Submitted By:

Anik Chandra

ID : 18101070

Submitted To:

Dr. Nasima Begum

Assistant Professor

Part-1:

Problem Statement:

1)Consider a list [A, B|C] = [a, [ ]]. Write a code segment to find out the values of A, B and
C?

Prolog Code: [A, B|C] = [a, []].

Sample Output:

```
?- [A, B|C] = [a, []].
A = a,
B = C, C = [].

?-
```

Problem Statement:

2)Consider the following rules for a List Searching:

is_in(Item,[Item|Rest]).
is_in(Item,[DisregardHead|Tail]):- is_in(Item,Tail).
Write a code segment to search carrot in a list of
[corn, cucumber, eggplant, carrot, broccoli].

Prolog Code:

```
is_in(carrot,[corn, cucumber, eggplant, carrot, broccol
i]).
```

Sample Output:

```
% e:/Anik/4.1 Semester/Artificial Intelligence Lab(404)/18101070part1
.pl compiled 0.00 sec, 2 clauses
?- is_in(carrot,[corn, cucumber, eggplant, carrot, broccoli]).
true
```

Problem Statement:

3)Given the types and body styles of a car:

types(Car, sedan) :-
has(Car, four_doors),
trunk(Car, traditional).
types(Car, coupe) :-
has(Car, two_doors),
trunk(Car, traditional),
roof(Car, solid).
types(Car, sports) :-
has(Car, two_doors),
price(Car, high).
Write a code segment to find out a specific car type with a query:
types(car, X) after
providing necessary facts.

Prolog Code:

```
has(sedan, four_doors).
trunk(sedan, traditional).

has(coupe,two_doors).
trunk(coupe,traditional).
roof(coupe,solid).

has(sports, two_doors).
price(sports, high).

types(Car, sedan) :-
has(Car, four_doors),
trunk(Car, traditional).

types(Car, coupe) :-
has(Car, two_doors),
trunk(Car, traditional),
roof(Car, solid).

types(Car, sports) :-
has(Car, two_doors),
price(Car, high).
```

Sample Output:

```
% e:/Anik/4.1 Semester/Artificial Intelligence Lab(404)/18101070part1
3.pl compiled 0.00 sec, -1 clauses
?- types(sports,X).
X = sports.

?- types(sedan,X).
X = sedan ;
false.

?- types(coupe,X).
X = coupe
```