**Artificial intelligence lab questions**

**SET (A)**

**Q1]**

**[1 mark for correct move steps (.25 each)**

**1 mark for correct canget facts.(.5 each)**

**0.5 marks for correct execution using trace. command.]**

move(state(middle,onbox,middle,hasnot), % before move

**grasp**, % grasp banana

state(middle,onbox,middle,has)). % after move

move(state(P,onfloor,P,H),

**climb**, % climb box

state(P,onbox,P,H)).

move(state(P1,onfloor,P1,H),

**push(P1,P2),** % push box form P1 to P2

state(P2,onfloor,P2,H)).

move(state(P1,onfloor,B,H),

**walk(P1,P2),** % walk form P1 to P2

state(P2,onfloor,B,H)).

canget(state(\_, \_, \_, has)). % Monkey already has it, goal state

canget(State1):- %do some work to get it

move(State1,Move,State2), %do something

canget(State2). %Get it now.

**Use trace command to see the steps on how monkey can get banana**

* **Walk -> climb -> push ->climb -> grasp**

**Q2]**

square :- **[1 mark]**

write( 'Enter the side of square '),

read( X),

area( X).

area(stop) :- !. **[0.5 Mark]**

area( N) :- **[1 Mark]**

C is N\*N,

write('Area of square with side '), write(N), write('is'),

write( C), nl,

square.

**Artificial intelligence lab questions**

**SET (B)**

**Q1] 0.5 marks for each right answer and solution**

female(pam). female(liz). female(pat). female(ann).

male(jim). male(bob). male(tom). male(peter).

Parent(pam,bob).parent(tom,bob).parent(tom,liz),parent(bob,ann).parent(bob,pat).

1. **mother(X,Y):- parent(X,Y),female(X).**

X= pam,Y= bob

1. **father(X,Y):- parent(X,Y),male(X).**

(X= tom,Y= bob/liz) and (X= bob, Y= ann/pat)

1. **haschild(X):- parent(X,\_).**

X=pam/tom/bob

1. **sister(X,Y):- parent(Z,X),parent(Z,Y),female(X),X\==Y.**

(X = liz, Y = bob) and (X = ann,Y = pat) and (X = pat,Y = ann).

1. **grandmother(X,Z):-mother(X,Y),parent(Y,Z).**

(X = pam, Y = ann) and (X = pam, Y = pat)

**Q2] [1 mark for proper input and processing**

**1 mark for correct execution of max and min.**

**0.5 marks for looping]**

minmax :- **[1 Mark]**

nl, write('A prolog program to compare two numbers'),

nl,

write('Enter first number: '),

read(X),

write('Enter second number: '),

read(Y),process(X,Y).

process(X,Y) :-

maxi(X,Y), mini(X,Y),minmax. % looping using minmax **[0.5 Marks]**

**[1 Mark]**

maxi(X,Y):- X>Y, write(X),write(' is greater than '),write(Y),nl.

maxi(X,Y):- X<Y, write(Y),write(' is greater than '),write(X),nl.

mini(X,Y):- X<Y, write(X),write(' is smaller than '),write(Y),nl.

mini(X,Y):- X>Y, write(Y),write(' is smaller than '),write(X),nl.