**1 Denote by** † **the basic AI problem of *an agent***

***acting intelligently in its environment*.**

**1.1 What is a Turing machine and what does it have to do**

**with** †**?**

• Environment is the Tape, it needs to halt

**1.2 What is the halting problem and how does it relate to** †**?**

• Whether a turing machine on a particular input will halt or not

• Undecidable in general

**1.3 What is the SAT problem and how is it related to** †**?**

• Trying to find an assignment to the variables satisfying an expression

• Might be the task the agent is trying to complete

• Boolean expressions are a way of expressing what it’s trying to complete

**1.4 What is the P vs NP problem, and how does it relate to**

**SAT?**

• Feasible computation

• Cobham’s theoerm

• SAT feasible if P = NP

• N allows for non determinism

**2 A Simple way in Prolog to search is**

search(Node) :- goal(node). search(Node) :- arc(Node, Next), goal(next)

**2.1 What is non-determinism? And how does it relate to**

**search?**

• There could be more than one next1

**2.2 Modify this search to do:**

**2.2.1 Bounded Depth First**

*depth first to a specific depth*

bs(Node, \_) :- goal(Node). bs(Node, s(X)) :- arc(Node, Next), bs(Next, X).

**2.2.2 Iterative Deepening**

*bounded depth fist until search succeeds*

iterSearch(Node) :- bound(Bd), bs(Node, Bd). bound(s(x)) :- bound(x).

**2.3 What are the ingredients for *A Star* Search?**

1. A cost on arcs 2. heuristic function on nodes indicating how close to goal node (minimum cost

path to goal node) 3. Frontier search: put at head of list the node with minimal F-Value

• *F*(*node*) := *Cost*(*node*) + *HeuristicV alue*(*node*)

fs([H|\_]) :- goal(H). fs([H|T]) :-

findall(X, arc(H, X), Children), addToFrontier(Children, T, New), fs(New).

**2.4 What does it mean For A-Star to be admissible?**

• If the search returns a solution, it returns an optimal solution (smallest cost).

• Minimal progress is made (within some epsilon), never overestimate cost.

**2.5 What are the ingredients of a Contstraint Satisfaction**

**Problem?**

1. Variables 2. Domain 3. Constraints

• 3 Color problem: Variables are nodes

• Domain is {*Red, Green, Blue*}3

2

**–** every node can take 3 colors, Red, Green and Blue **–** *m* = 3

• If there’s an arc between nodes, the colors must be different

**2.6 What is Generate and test?**

• Instantiate all of the variables before testing the constraints

member(X1, D1). % -- member(X2, D2). % | generation ... % | part member(Xn, Dn). % --

test(X1, Xn) % only test after instantiating

**3 Consider the knowlege base**

false :- p. false :- a, b. p :- b,c. p :- q. a :- r, s. b.**3.1 What are *Integrity Constraints*?**

• A rule in the KB where head is false

**–** Horn clause: clause with at most 1 positive literal

3

**3.2 Suppose** q**,** r**,** s **were assumable. What are the conflicts?**

**The Minimal conflicts?**

Find minimal conflicts by repeated substitution

falsep

a, b

qb, c

r, s, b

**3.3 What’s the complete knowlege assumption (CKA)?**

• Only atoms that are true are ones we can prove. If we can’t prove it we take it as false.

**3.4 What does non monotonicity with respect to inference**

**systems mean?**

• *KB* ⊣ *C* =⇒ *KB* ∪ {*a*} ⊣ *C*

**3.5 What does it mean for a KB to be Consistent?**

• Consistent KB =⇒ ∃ a model for the knowlege base

**–** Model: Interpretation where all clauses are true

4