Modern University
For Technology & Information
Faculty of Computers & Information

Academic Year: 2018/2019
Semester: Spring
Specialization: CS
Time: 3 Hours
Total Mark: 60

Course: (CS 431) Artificial Intelligence

Examiners: Prof.Dr.Hesham El-Deeb
Prof.Dr.Hafez. Sayed Abdel Wahab

Questions for Final Written Examination

Number of Questions (5)	**Number of Pages (2)**

Question 1: [12 Marks]

Monkey plan problem is a famous example of planning applications. It could be stated as: "A monkey is in a lab. The monkey wants some bananas. There are three locations in the lab – locations A, B and C. The monkey is at location A. There is a box in location C. There are some bananas in location B, but they are hanging from the ceiling. The monkey needs the box to get to the bananas."

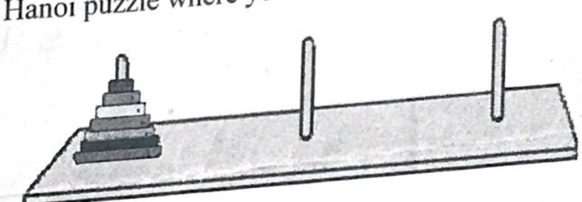
(a) State four practical examples of Planning Applications. **[4 Marks]**

(b) Clarify how to generate a plan in the AI field. **[4 Marks]**

(c) Write STRIPS program to solve the Monkey plan problem. **[4 Marks]**

Question 2: [12 Marks]

Consider the seven-disk Towers of Hanoi puzzle where you have seven disks and three rods.



(a) Describe the basic concepts of the Towers of Hanoi puzzle. **[4 Marks]**

(b) How many steps to solve this specific problem? **[4 Marks]**

(c) Solve only the first eight steps in descriptive and graphical explanation. **[4 Marks]**

Question 3: [12 Marks]

A* search is one of the most popular informed search algorithms. Consider the following graph:

No:------------ Date:------------

GIS

GPS تطبيقات
Source & dest

Lect (4) Informed Search Algorithm

4.1 Best first search 4.2 Greedy best first search 4.3 A* search
enhanced greedy best first search

الأفضل على حسب مضمون الهدف

** Best first search:

→ it's search algorithm which explores a graph by expanding the most promising node chosen according to some rule.

→ it also described as simplest informed search algorithm, which estimate the promise node. n by a"
* heuristics evaluation function f(n) which may depend on:
  1. the description of n
  2. the description of goal.
  3. the information gathered by search up to that point.
  4. knowledge about the problem domain.

→ often used for path finding in Combinatorial search.

→ it's typically implemented using a priority queue.

** Algorithm:

1. start with open containing just the initial node (state).
2. until goal state is found or there is no node left on open do:
   * pick the best node on open
   * generate it's successors
   for each successor do:
   → if it's not in open, add it to open & record it to parent. (new node)
   → old node change the parent, if this new path is better than previous one.

** Properities :-

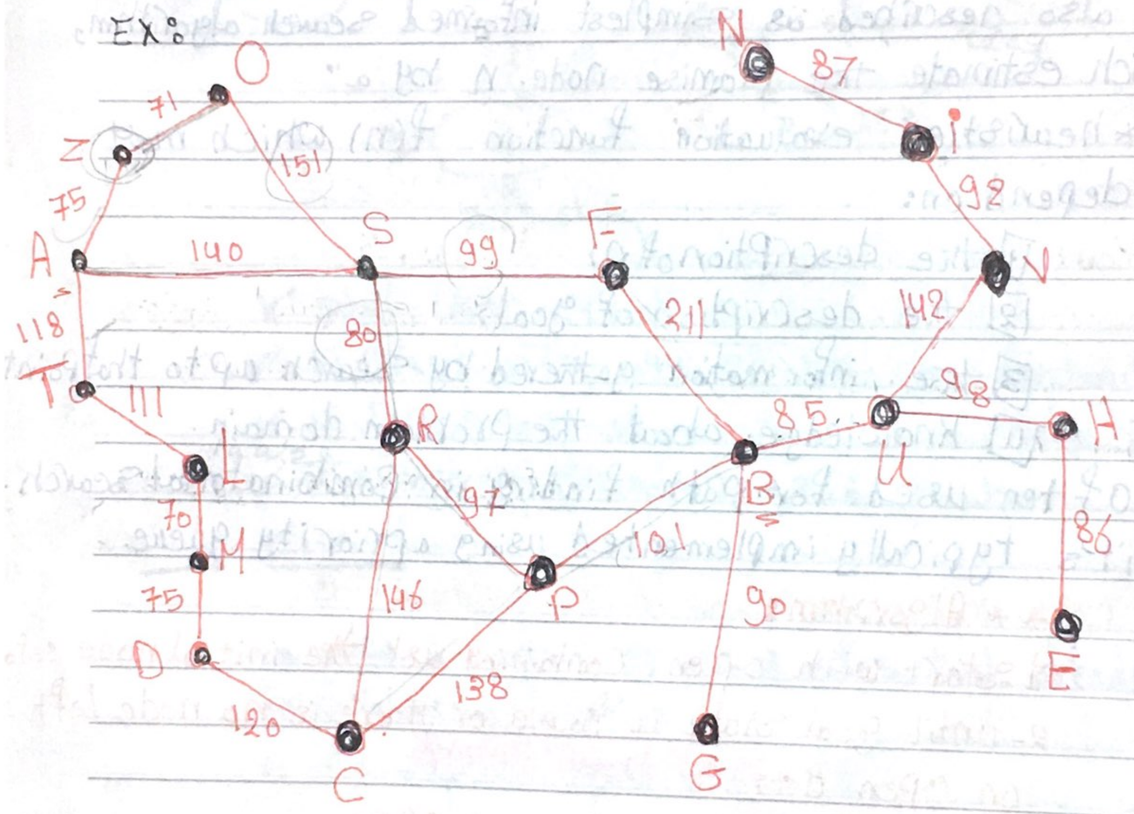
→ Not Complete : Because it may follow infinite path if heuristics rates state on the best option.

→ worest case time complexity is also $O(b^m)$ where m is the maximum depth, b is the branching factor.

→ Space complexity is also $O(bm)$.

straight line distance

Ex:



A	366
B	0
C	160
D	242
E	161
F	176
G	77
H	151
I	226
L	244
M	241
N	234
O	380
P	10
R	193
S	253
T	329
U	80
V	199
Z	374

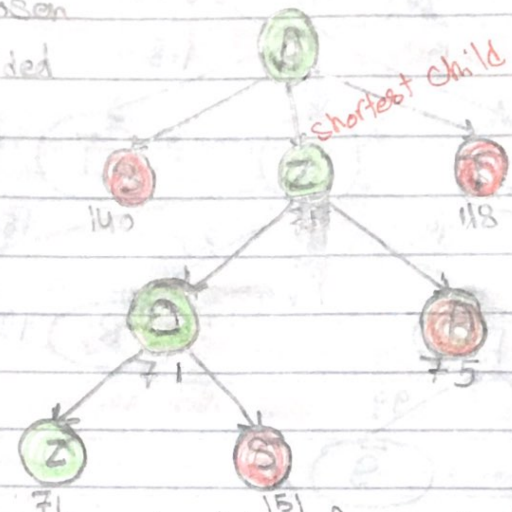
→ Dis advantage:
Doesn't found the shortest path to goal (through all Rimnicu) Since, because it only forcused on the cost remaining rather than total cost.

No:------------ Date:------------

Solving according the Best first from A to B

Choosen

avoided



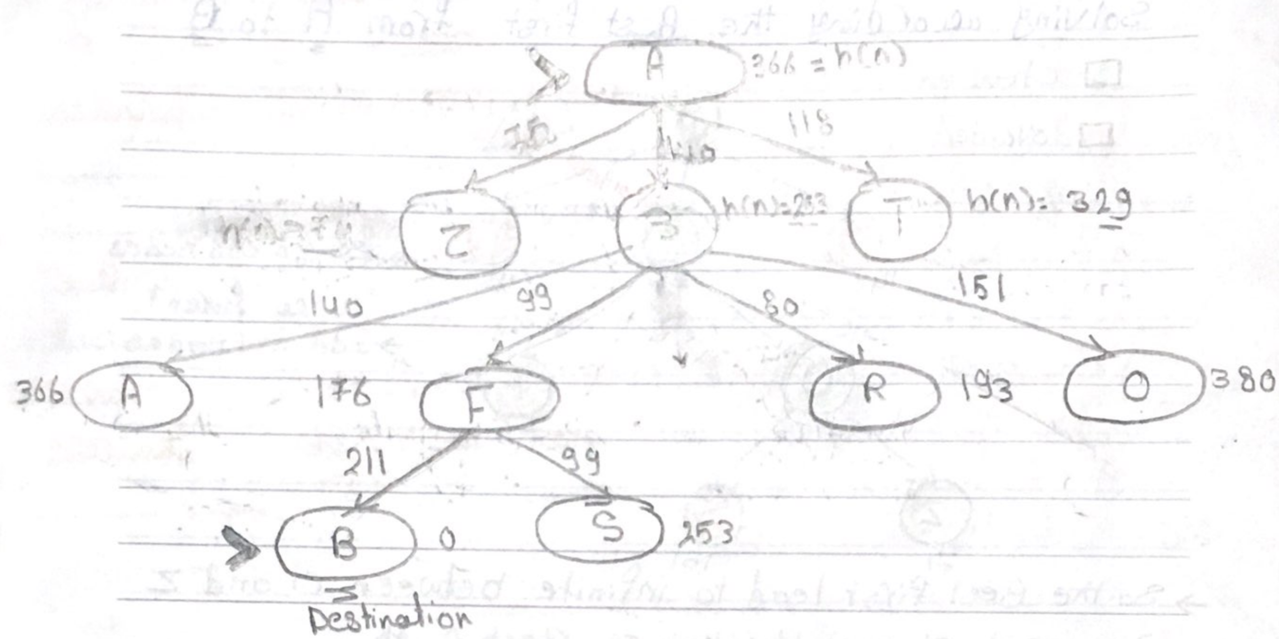
→ assign old nodes to new Parent
→ add new nodes

→ So the Best first lead to infinite between O and Z Because it choose only the shortest path
→ lead to stop & doesn't find the shortest path Because it focuse only on the cost remaining rather than total cost.

الأقرب للهدف وفقاً لمفتاح الخريطة

** Greedy best first search: تقليل عدد اللينكات بقدر الإمكان
→ the path that judged to be closer to solution are extended first. وسع
→ Evaluation function
$f(n) = h(n)$ = estimate of cost from n to goal
→ Proberities:
Complete: No, can get stuck in loops عالق
Time $O(b^m)$: may be good of heuristics can give improvement. ارشادى
Space $O(b^m)$: Keeps all nodes in memory
optimal: No,

No:------------ Date:------------



✓ total Links = 3 Link

✓ in graph = 140 + 99 + 211 = 450 Km = Distance on map

✓ in Direct distination = 366 Km = straight line distance

✓ the lost in Km = 450 − 366 = 84 Km

A* search , enhanced greedy Best first search)

→ Idea : avoid expanding paths that already expensive

→ Evaluation function

$f(n) = g(n) + h(n)$

↳ cost so far reach n (accumulative) تراكمية / تكاملية

↳ estimated Cost from "n" to goal مقدار

↳ estimated Cost of path through n to goal. مقدار

→ properities:

Complete : Yes

Time : Exponential

Space : Keeps all nods in memory

Optimal : Yes

No:------------ Date:------------

total No of links = 4 links
Distance on map = 151 + 80 + 97 + 101 = 429 Km
Straight line = 380 Km
Diff = 380 - 429 = 49 Km

	Greedy	A* Algorithm
Complete	No, can get stuck in loops	Yes, can give solution
time	may be good & can give imp	Exponential
Space	Keeps all nodes in memory	Keep all nodes in memory
optimal	No, Because there is mor	Yes, Shortest Path

No:------------ Date:------------

"Lect 3"

→ Search sterategy is defined by : the order of node expansion.

→ Strategies are evaluated along the following dimensions:

[1] Completeness: finding asolution if one is exist.
[2] time Complexity: number of nodes generated,
[3] Space complexity: max number of nodes in memory.
[4] optimality: does it always find aleast cost solution

→ Time & Space Complexity (for Comparison purposes) are measured in terms of:

b: maximum branching factor of search tree.
d: depth of the least Cost solution.
m: maximum depth of the state space.

Breadth first search: "BFS"

→ it's uniformed search method that aims to expand & examine all nodes of graph.

unformed

→ dis adv:

① * it's exhaustively search that search is run over all graph or sequence untill find the goal.

② * it doesn't use aheuristics

→ technique:

* All Child node obtained & added to a FIFo queue.
* nodes that not examined for their neighbors are placed in some Container Called "open"
* nodes that has been examined are placed in Container "closed".

No:------------ Date:------------

BFS Algorithm:

1- put the root node on the queue.

2- pull the node from the begining of queue & examine it.

* if node is found return the result
* other wise push all the (un examined successors) into the end of queue.

3- if queue is empty, every node has been examined, Quite the search and return not found.

advantage or uses, for solving problems in graph theory:

(1) find all Components in a graph. ✓

(2) Copying Collections ✓

(3) finding the shortest path between nodes ✓

Assign ment #4:

(a) given that: $b=10$, $m=9$, $d=5$, $L=3$

Complete the next table, according to the rules for the three types of search algorithm.

	Depth first	Depth Limited	iterative Deeping
Space Complexity			
Time Complexity			

Depth first → time Complexity $O(b^m)$
→ space Complexity $O(bm)$

Depth limit → time Com $O(b^L)$
→ Space Com $O(bL)$

iterative Deepening → t Com
→ S Com

YASSIN

No:---------- Date:----------

Depth first:

Time Complexity:

$b = 10$, $m = 9$

$O(b^m) = (m+1)b^0 + (m)b^1 + (m-1)b^2 + (m-2)b^3$
$(m-3)b^4 + (m-4)b^5 + (m-5)b^6 + (m-6)b^7$
$(m-7)b^8 + (m-8)b^9$

$= (9+1)(10)^0 + 9(10)^1 + (9-1)(10)^2 + (9-2)(10)^3$
$(9-3)(10)^4 + (9-4)(10)^5 + (9-5)(10)^6 + (9-6)(10)^7$
$(9-7)(10)^8 + (9-8)(10)^9$
$= (10)(10)^0 + 9(10) + (8)(10)^2 + (7)(10)^3$
$+ 6(10)^4 + 5(10)^5 + (4)(10)^6 + (3)10^7 + (2)(10)^8$
$+ 1(10)^9 =$
$10 + 90 + 800 + 7000 + 60000 + 500000$
$4000000 + 30000000 + 200000000$
$+ 1000000000 = 1234567900$

Space Complexity:

$O(bm) = (m+1)b(0) + (m-1)b(1) + (m-1)(b)(2) \ldots$
$\ldots (m-8)*(b)*(9)$
$= (9+1)(10)(0) + (9)(10)(1) + (8)(10)(2)$
$+ (7)(10)(3) + (6)(10)(4) + (5)(10)(5) +$
$(4)(10)(6) + (3)(10)(7) + (2)(10)(8)$
$+ (1)(10)(9) =$
$= 0 + 90 + 160 + 210 + 240 + 250 + 240 +$
$210 + 160 + 90 = 1650$

No: Date:

Depth first: $b = 10$, $m = 9$

Time Complexity:

$O(b^m) = (m+1)b^0 + (m)\,b^1 + (m-1)b^2 + (m-2)b^3$

$(m-3)b^4 + (m-4)b^5 + (m-5)b^6 + (m-6)b^7$

$(m-7)b^8 + (m-8)\,b^9$

$= (9+1)(10)^0 + 9\,(10)^1 + (9-1)(10)^2 + (9-2)(10)^3$

$(9-3)(10)^4 + (9-4)(10)^5 + (9-5)(10)^6 + (9-6)(10)^7$

$(9-7)(10)^8 + (9-8)(10)^9$

$= (10)(10)^0 + 9(10) + (8)(10)^2 + (7)(10)^3$

$+ 6(10)^4 + 5(10)^5 + (4)(10)^6 + (3)10^7 + (2)(10)^8$

$+ 1(10)^9 =$

$10 + 90 + 800 + 7000 + 6000 + 500000$

$4000000 + 30000000 + 200000000$

$+ 1000000000 = 1234567900$

Space Complexity:

$O(bm) = (m+1)b(0) + (m)\,b(1) + (m-1)(b)(2) \_\_$

$\_\_ \_\_ (m-8)*(b)*(9)$

$= (9+1)(10)(0) + (9)(10)(1) + (8)(10)(2)$

$+ (7)(10)(3) + (6)(10)(4) + (5)(10)(5) +$

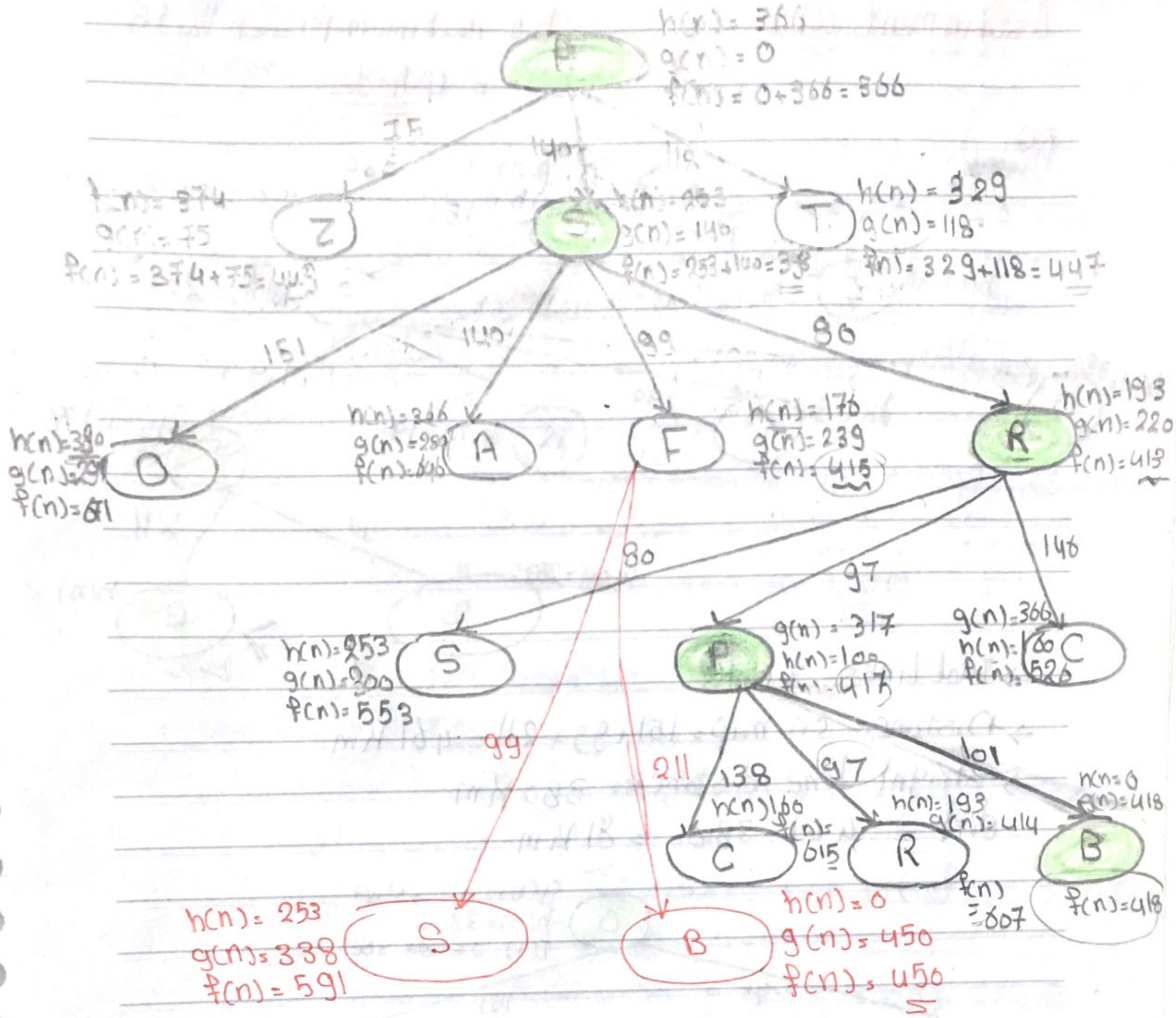
$(4)(10)(6) + (3)(10)(7) + (2)(10)(8)$

$+ (1)(10)(9) =$

$= 0 + 90 + 160 + 210 + 240 + 250 + 240 +$

$210 + 160 + 90 = 1650$

optimal اقصر طريق



the Choosen Path is to be follow green one

# of Links = 4 Links

Distance from map = 101 + 97 + 80 + 140 = 418 km

stright line distance = 366 Km

Diff Between distance = 418 - 366 = 52 K.

m: maximum depth of the state space = [illegible] levels

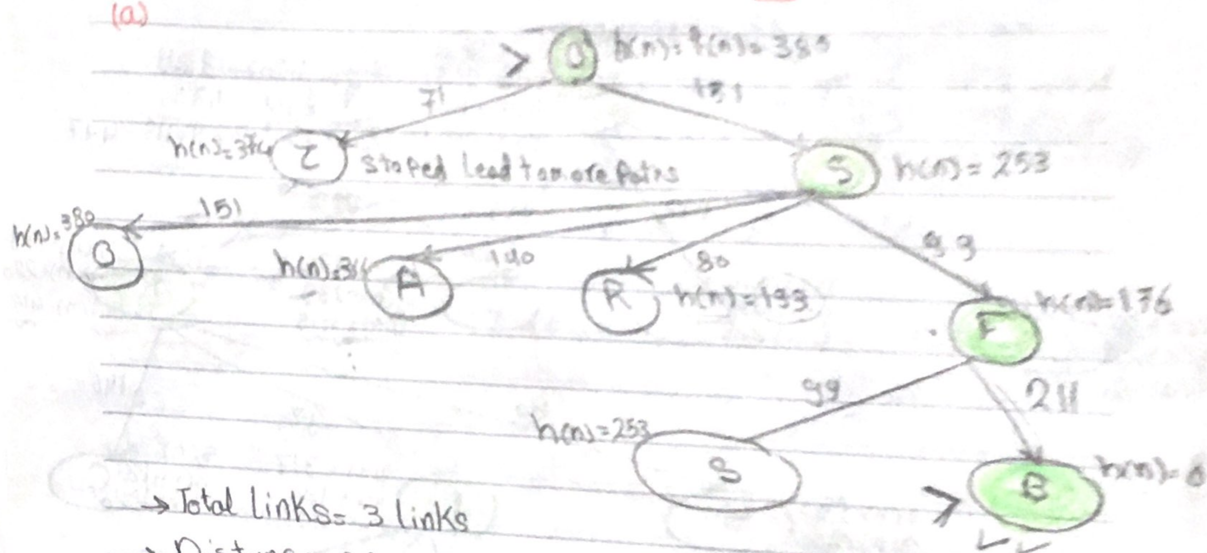
d: depth of least cost solution = [illegible] levels

b: maximum Branch factor

= 4 nodes

Assignment (6):

(a)



→ Total links = 3 links

→ Distance on map = 151 + 99 + 211 = 461 Km

→ Strignt line distance = 380 Km

diff = 461 - 380 = 81 Km

No:--------

Date:----------

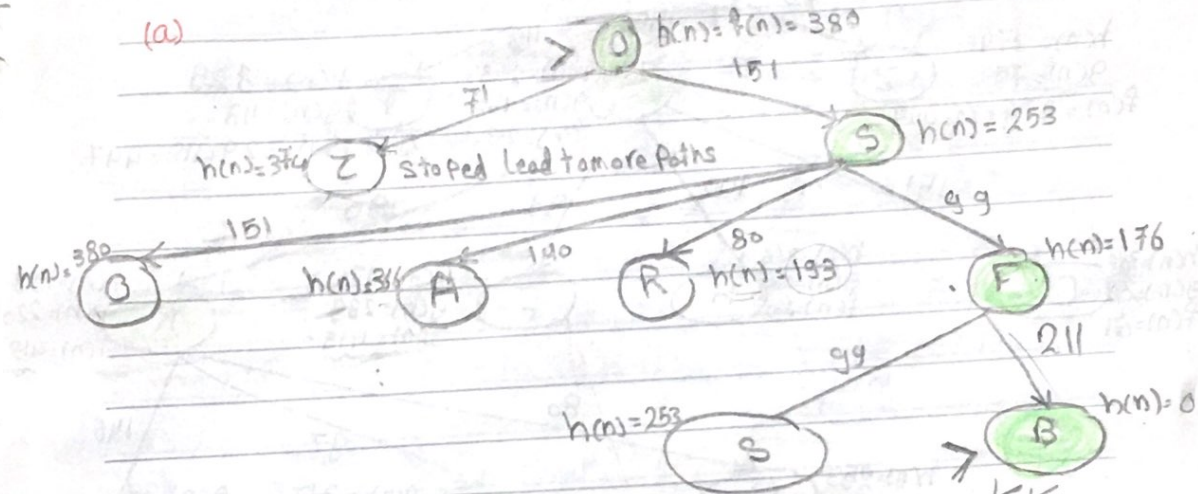
m: maximum depth of the state space = 4 levels

d: depth of least cost solution = 4 levels

b: maximum Branch factor = 4 nodes

Assignment (6):

(a)

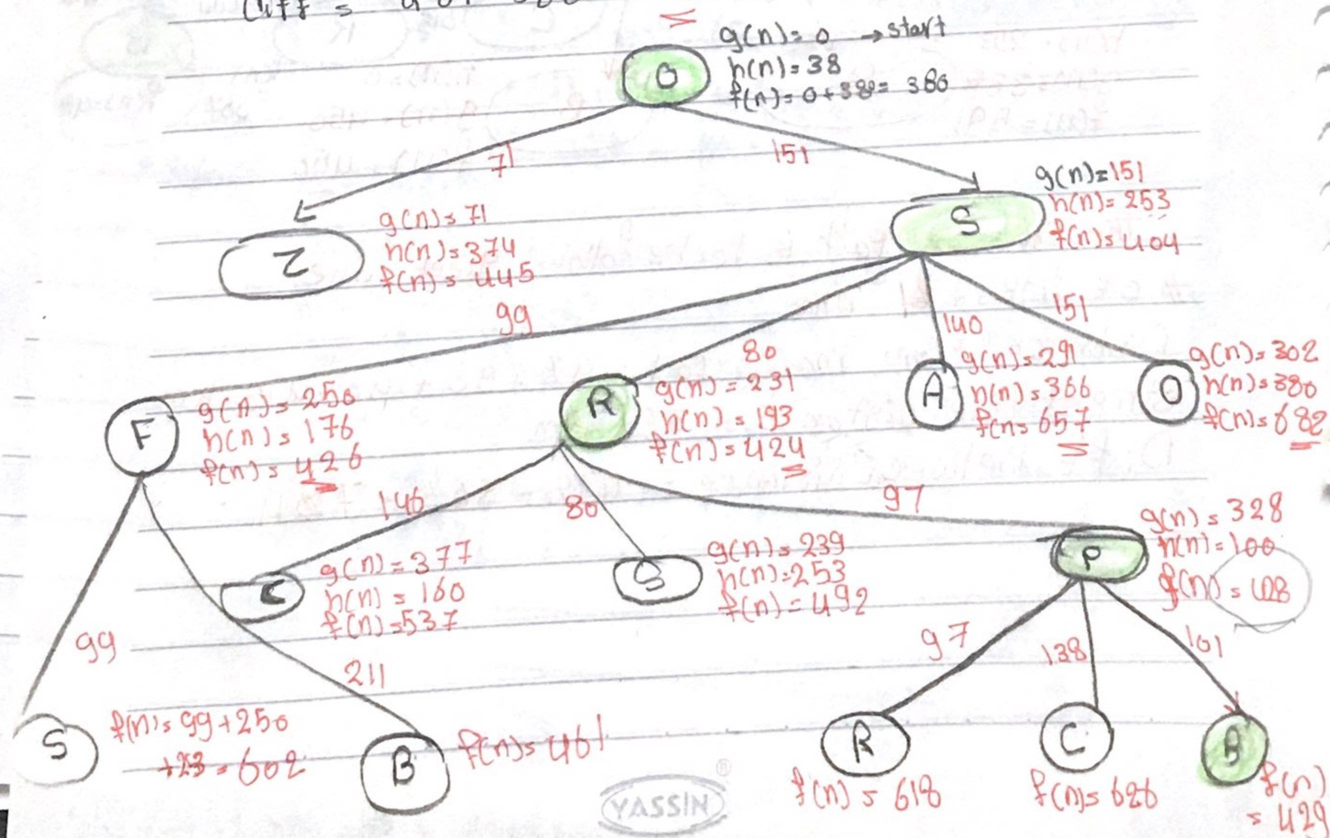


→ Total links = 3 links

→ Distance on map = 151 + 99 + 211 = 461 Km

→ Stright line distance = 380 Km

diff = 461 - 380 = 81 Km



No:------------ Date:------------

Depth limited: b=10, L=3

* time Complexity

$O(b^L) = (L+1)\ b^0 + L\ b^1 + (L-1)\ b^2 + (L-2) b^3$

$= (3+1)(10)^0 + (3)(10)^1 + (3-1)(10)^2 + (3-2)10^3$

$= 4*1 + 30 + 200 + 1000 = \underline{1234}$

* Space Complexity

$O(bL) = (L+1)\ b(0) + L\ b(1) + (L-1) b(2) + (L-2)\ b(3)$

$= (3+1)(10)(0) + (3)(10)(1) + (3-1)(10)(2) + (3-2)(10)(3)$

$= 0 + 30 + 40 + 30 = \boxed{100}$

Iterative deepening search: b=10, d=5

*time Complexity

$O(b^d) = (d+1) b^0 + (d)\ b^1 + (d-1) b^2 + (d-2) b^3 + (d-3) b^4$
$(d-4) b^5$

$= (5+1)(10)^0 + (5)(10)^1 + (5-1)(10)^2 + (5-2)(10)^3 + (5-3)(10)^4 + (5-4)(10)^5$

$= (6)(1) + 50 + 400 + 3000 + 20000 + 100000 = \boxed{123456}$

* Space Complexity

$O(bd) = (d+1)\ b(0) + (d)(b)(1) + (d-1)(b)(2) + (d-2) b(3) + (d-3) b(4) + (d-4) b(5)$

$= (5+1)(10)(0) + (5)(10)(1) + (5-1)(10)(2) + (5-2)(10)(3) + (5-3)(10)(4) + (5-4)(10)(5)$

$= \cancel{(6)}\ 0 + 50 + 80 + 90 + 80 + 50$

$= \boxed{350}$

YASSIN

→ Iterative deepening search vs depth limited search

while : time Complexity : is the number of nodes that generated

so in depth limited search # of nodes generated to depth "d" with branching factor b

$$N_{DLS} = b^0 + b^1 + b^2 \dots b^d$$

branch factor

depth of the least cost solution [illegible]

So b, d is effecting value in Calculating the space & time Complexity

Modern University
For Technology & Information
Faculty of Computers & Information

Academic Year: 2017/2018
Semester: Spring
Specialization: CS
Time: 3 Hours
Total Mark: 60

Course: (CS 431) Artificial Intelligence
Examiners: Prof.Dr.Hesham El-Deeb
Prof.Dr.Hafez. Sayed Abdel Wahab

Questions for Final Written Examination

Number of Questions (5)	Number of Pages (2)

Question 1: [12 Marks]

The A* search is one of the informed search algorithms.Consider the following example in **fig.1**

(a) Clarify the concept of the evaluation function of the A* algorithm. **[6 Marks]**
(b) Trace and explain the computations in fig.1 using the A* search algorithm. **[6 Marks]**

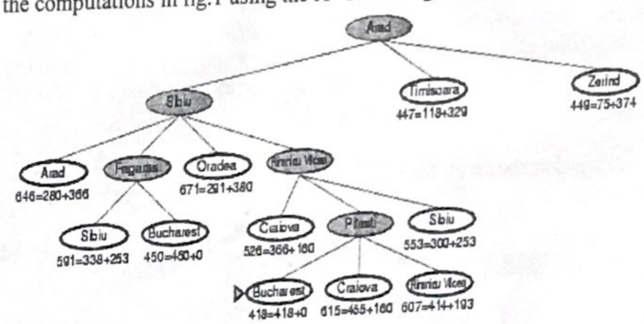


fig.1

Question 2: [12 Marks]

Consider the Map Coloring problem. It is required to color **Australia** map in **fig.2** using three colors **[Red]**, **[Green] and [Blue]** such that adjacent regions have different colors.

Australia

fig.2

(a) State the variables, domains and constraints for fig.2. **[4 Marks]**
(b) Draw the Initial State Representation for the above map. **[4 Marks]**
(c) Propose and explain your solution to color Australia map in fig.2. **[4 Marks]**

Question 3: [12 Marks]

Consider the Depth-limited search and the Iterative deepening search algorithms, where time and space complexity, for comparison purpose; are measured in terms of four parameters: **b** (maximum branching factor of the search tree) ,**d** (depth of the least-cost solution) , **m** (maximum depth of the state space) and ***l*** (depth limit, i.e., nodes at depth *l* have no successors)

	Depth Limited Search	**Iterative Deepening Search**
Space Complexity	*O(bl)*	*O(bd)*
Time Complexity	$O(b^l)$	$O(b^d)$

(a) Given that the search space is described as follows: b = 10, m = 9, d= 5, *l* = 3. Compute the space and time complexity for the above two search algorithms. **[4 Marks]**
(b) Re-compute the same table, if the values changed to be: b = 5, m = 9, d= 10, *l* = 3. **[4 Marks]**
(c) Based on your results in (a) and (b).Which search algorithm is better than the other? **[4 Marks]**

Question 4: [12 Marks]

Consider the following blocks arranged as shown in, fig.3

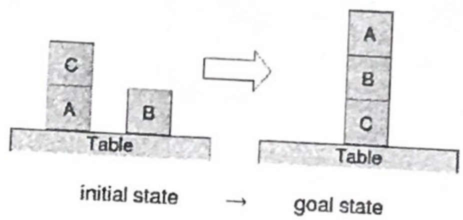


fig.3

(a) Explain the concept of the Blocks World (Sussman Anomaly) problem. **[4 Marks]**
(b) Describe the initial state and goal state in fig.3 **[4 Marks]**
(c) Staring from the initial state, write a STRIPS program to reach to the goal state. **[4 Marks]**

Question 5: [12 Marks]

Crypt arithmetic is a type of a mathematical game consisting of a group of mathematical equations among unknown numbers, whose digits are represented by letters. The goal is to identify the value of each letter.
(a) State the formal mathematical definition of the constraint satisfaction Problem. **[4 Marks]**
(b) Solve the following Crypt arithmetic problem: **[8 Marks]**

112

```
    8 7
    N O
  9 0 8
  G U N
+   N O
    8 7
---------
H U N T
1 0 8 2
```

Good Luck

Final 2016 – 2017

Modern University for Technology & Information — Academic year: 2016/2017

Faculty: Computers and Information — Semester: Fall 2016

Course: (CS431) Artificial Intelligence — Specialization: Computer Science

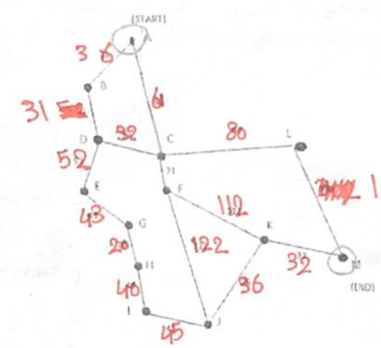
Examiner: Prof. Dr. Hesham El-Deeb — Time: 3 Hours

Questions for Final Written Examination

Number of Questions: 4	**Number of Pages : 2**

Question 1: [15 Marks]

Consider the following map connecting a group of towns A,B,D,C,L,E,F,K,M,G,H,I,J.



The A* algorithm could be used to work out a route from town X to any town Y by using the following cost functions:

- g(n) = The cost of each move as the distance between each town (shown on map).
- h(n) = The Straight Line Distance between any town and town M. These distances are given in the table below.

Straight Line Distance to M form each town in Kilometers

A	223	E	165	I	100	M	0		
B	222	F	136	J	60				
C	166	G	122	K	32				
D	192	H	111	L	102				

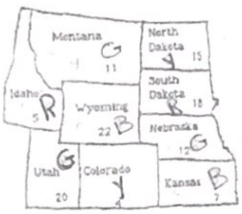
(a) Show the A* algorithm search tree with the cost function at each node. **[10 Marks]**

(b) State the order in which the nodes were expanded. **[3 Marks]**

(c) State the route that is taken, and compute the total cost. **[2 Mark]** informed search Algorithm — BFA Greedy A* Algorithm

Question 2: [15 Marks]

Consider the problem of Map colouring as a constraint satisfaction problem. It is required to colour the following map using four colors: Red, Yellow, Green and Blue such that adjacent regions have different colors.



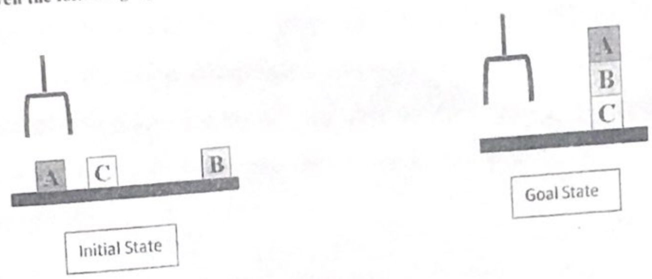
(a) State the variables, domains and constraints for the previous map .[5 Marks]

(b) Using the **Backtracking search algorithm**, propose your solution to color the previous map. [10 Marks]

Question 3: [15 Marks]

The Sussman Anomaly or blocks world is a problem in artificial intelligence (AI), first described by Gerald Sussman that illustrates a weakness of noninterleaved planning algorithms.

Given the following figure:



Given → world → goal — initial → action

→ ach 2 Subgoal 2

(a) Give four examples for the planning applications in AI? [3 Marks]

(b) What is the relation between the planning problems in AI and the blocks world? [3 Marks]

(c) Describe the initial state and the goal state in blocks world terminology.[6 Marks]

(d) Explain your plan to reach the goal state starting from the initial state.[3 Marks]

Question 4: [15 Marks]

Crypt arithmetic is a type of mathematical game consisting of a mathematical equation among unknown numbers, whose digits are represented by letters. The goal is to identify the value of each letter.

(a) State the formal mathematical definition of the constraint satisfaction Problem. [5 Marks]

(b) Solve the following Crypt arithmetic problem: [10 Marks]

```
   EAT
+ THAT
------------
 APPLE
```

Answer:

Q1. Assignment 6

Q2. Lec 5 "Example 4"

Q3. Lec 6 "Example 2"

Q4. Assignment 8 – Example 5

Modern University

For Technology & Information

Faculty of Computers & Information

Course: **(Artificial Intelligence)**

Examiner: **Dr. Hesham El-Deeb**

Academic year: 2017 / 2018

Semester: Spring

Specialization: Computer Science

Time: **90** minutes

Questions for Midterm Examination

Number of Questions: 3	Number of Pages: 2

Question 1: [7 Marks]

Buyer agents travel around a network (e.g. the internet) retrieving information about goods and services. These agents, also known as 'shopping bots', work very efficiently for buying products such as CDs, books, electronic components, and other one-size-fits-all products. Buyer agents are typically optimized to allow for digital payment services used in e-commerce and traditional businesses. Amazon.com, is a shopping bot example that displays a list of similar books the customer may be tempted to buy.

(a) Describe the (Performance measure, Environment, Actuators and Sensors) for this agent. **(2 Marks)**

(b) Do you think that this agent could be intelligent? Why? **(2 Marks)**

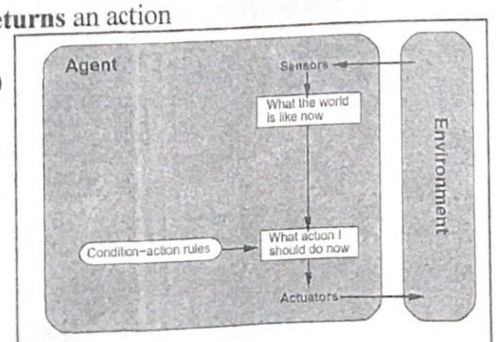
(c) If this agent is designed based on the condition-action rule: if condition then action rule. **(2 Marks)**

(d) Do you think that this agent is model-based reflex agent? Why? **(1 Marks)**

Question 2: [7 Marks]

Assume you have a **simple reflex agent** behave as follows:

```
function REFLEX_VACUUM_AGENT( percept ) returns an action
    (location, status) = UPDATE_STATE (percept)
    if status = DIRTY then return SUCK;
      else if location = A then return RIGHT;
          else if location = B then return LEFT;
```



(a) Analyze the relation between the **code** and the **design** of this agent. **(2 Marks)**

(b) Determine the Task Environment Description for this agent in form of (**PEAS**): Performance measure, Environment, Actuators, Sensors. **(2 Marks)**

(c) Add your convenient **modification** to the design and the code to accomplish **Model-based** reflex Vacuum agents. **(2 Marks)**

(d) Is this environment of this agent is **fully observable** or **partially observable**? Why? **(1 Marks)**

Question 3: [6 Marks]

Given the following PEAS for the **automated taxi driver** agent,

***Performance measure*:** safe, fast, legal, comfortable, maximize profits

***Environment*:** roads, other traffic, pedestrians, customers

***Actuators*:** steering, accelerator, brake, signal, horn

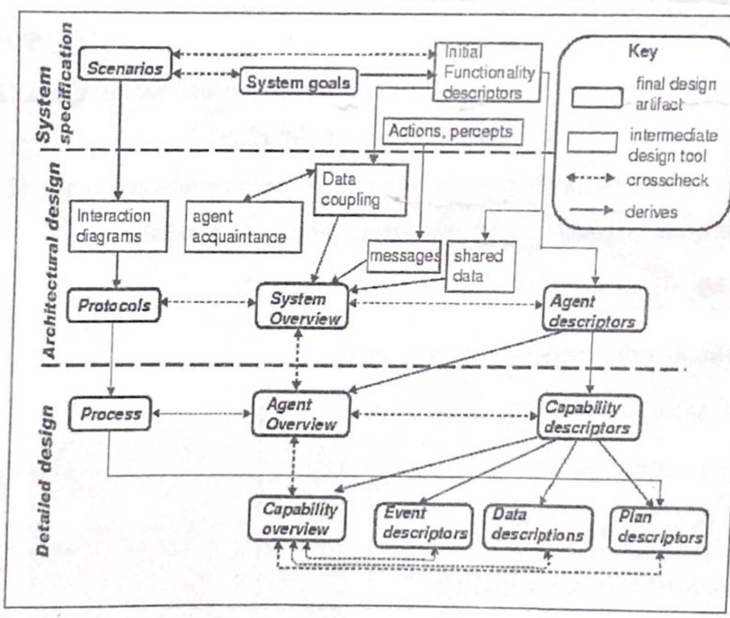
***Sensors*:** cameras, sonar, speedometer, GPS.

Describe your

(a) System specification. **(2 Marks)**

(b) Architecture design. **(2 Marks)**

(c) Detailed design for **one scenario** of the **automated taxi driver agent.** **(2 Marks)**



Good luck

Modern University

For Technology & Information

Faculty of Computers & Information

Course: **(Artificial Intelligence)**

Examiner: **Dr. Hesham El-Deeb**

Academic year: 2018 / 2019

Semester: Spring

Specialization: Computer Science

Time: **60** minutes

Questions for Midterm Examination

Number of Questions: **2**	Number of Pages: **2**

Question 1: [10 Marks]

The robot could be defined as: "A reprogrammable, multifunctional manipulator designed to move materials, parts, tools, or specialized devices through various programmed motions for the performance of a variety of tasks." In the essence of the practical case of **line Follower Robot** used for **Tour guides in museums:**

(a) What are these robots good at? **(1 Mark)**

(b) What other applications would you give to this robots? **(2 Mark)**

(c) What does building these robots teach us about humans? **(1 Mark)**

(d) How do these robots move? **(1 Mark)**

(e) What sensors might these robots have? **(2 Mark)**

(f) What are some problems with control of these robot actions? **(1 Mark)**

(g) How do you measures of performance of such robot? **(2 Mark)**

Question 2: [10 Marks]

Given the PEAS for the **automated taxi driver** agent as an example of the Unmanned Ground Viechle: ***Performance measure*:** safe, fast, legal, comfortable, maximize profits. **Environment:** roads, other traffic, pedestrians, customers. **Actuators:** steering, accelerator, brake, signal, horn. ***Sensors*:** cameras, sonar, speedometer, GPS.

(a) Explain the functionality of the general block diagram given in Fig.1 **(3 Marks)**

(b) Select **your own scenario** and describe your proposed software processes and hardware based on Fig.1 **(4 Marks)**

(c) Do you think this application uses the concept of **AI reasoning**? Why? **(3 Marks)**

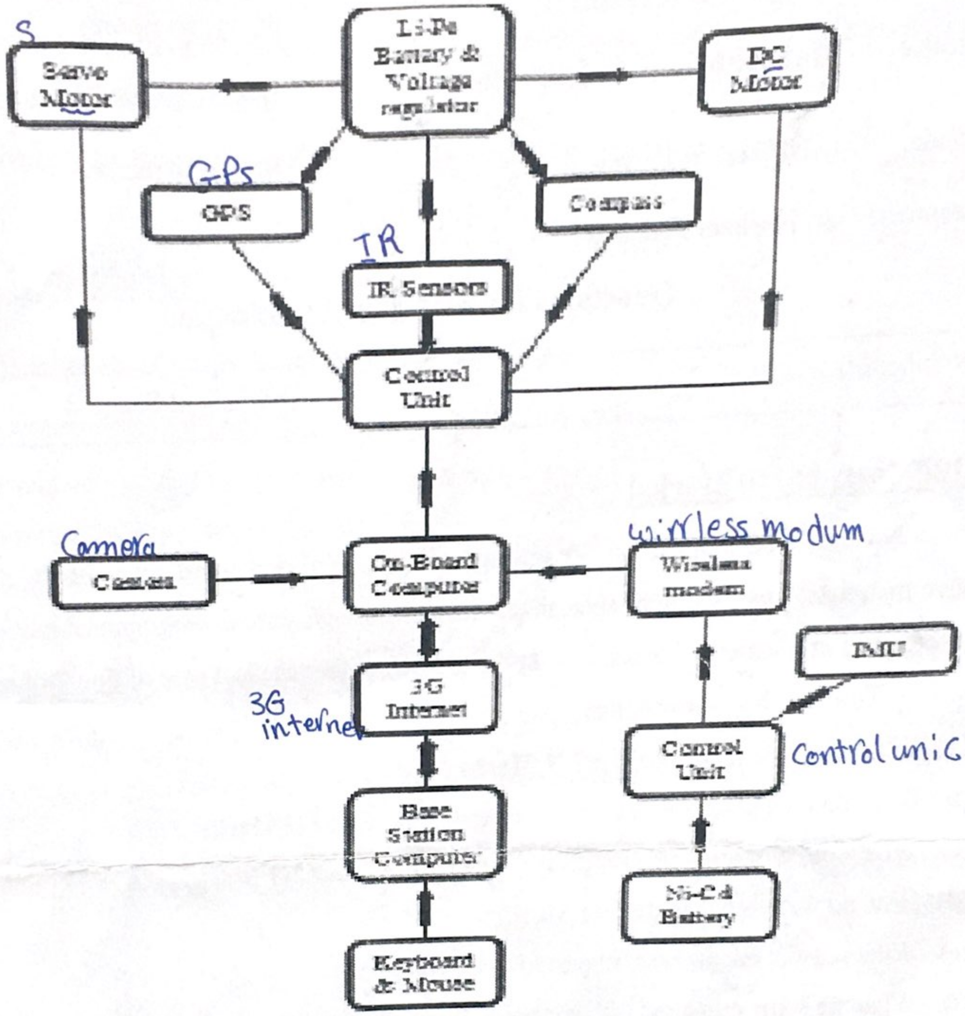


Fig.1 General Block Diagram of the automated taxi driver

Good Luck

AI – Final

Lec 3: Solving problems by searching

- Problem-solving agents:

```
function SIMPLE-PROBLEM-SOLVING-AGENT(percept) returns an action
   static: seq, an action sequence, initially empty
           state, some description of the current world state
           goal, a goal, initially null
           problem, a problem formulation

   state ← UPDATE-STATE(state, percept)
   if seq is empty then do
        goal ← FORMULATE-GOAL(state)
        problem ← FORMULATE-PROBLEM(state, goal)
        seq ← SEARCH(problem)
   action ← FIRST(seq)
   seq ← REST(seq)
   return action
```

- Problem types:
  - Deterministic, fully observable → single-state problem
    - Agent knows exactly which state it will be in; solution is a sequence
  - Deterministic, non-observable → sensor less problem (conformant problem)
    - Agent may have no idea where it is; solution is a sequence
  - Nondeterministic and/or partially observable → incident problem
    - percepts provide new information about current state
    - often interleave search, execution
  - Unknown state space → exploration problem

- Single-state Problem Formulation:
  - A problem is defined by four items:
    - initial state
    - actions or successor function S(x) = set of action–state pairs
    - goal test can be
      explicit, e.g., x = ""
      implicit, e.g., Checkmate(x)
    - path cost (additive)
      e.g., sum of distances, number of actions executed.

A solution is a sequence of actions leading from the initial state to a goal state