**1. AI approaches**

**a) Briefly summarize the major approaches in AI. You have to provide a discussion on the main focus of each approach. (2 marks)**

There are four main approaches in AI, acting humanly, thinking humanly, thinking rationally and acting rationally. Acting the humanly is the models of human behaviours, also known as the truing test approaches. At what point can we distinguish we can tell whether the it is a human or AI making the decision. Where it is the study of “how to make computers do things at which, at the moment, people are better.” (Rich and Knight, 1991)

The second approaches are thinking humanly, it is the effort of making computers “think with minds, in full and literal sense, what that it means is that when a computer thinks as a human, it requires a task that require intelligence to solve problem. Instead of one particular repetitive procedure to solve a certain task, computer must be able to determine a way to think like a human to solve complex problems. Hence you must define a model which detect and perceived technique to achieve goals through similar set of human circumstance, such as goals, need of resource, and environment condition.

The third approach is thinking rationally, which is an approach where computers must think rationally when solving a certain goal based of the recorded behaviours which creates a guideline of how it should interact with an environment through from its perceived data. Thus, the main objective of this approach to make computer be able to solve problem logically as possible.

Last approached that makes an AI, is acting rationally. It is an approach of how humans act in given situation. Computers must be able to reflect and respond to a specific scenario through an effective and efficient technique.

**b) What is the difference between weak AI and Strong AI?**

Weak AI is an artificial intelligence that concentrate on narrow task. Weak artificial intelligence has limited predefined functions. Weak AI is system that can only act it thinks, it uses a set of instructions that solve a narrow problem. For instance, Siri is a Weak AI, because it can only answer based on the set of answer as from the database. Siri seems intelligent, it can hold conversation with users, but actually operates a very narrow set of provided answer from the database. Thus, weak artificial intelligence is an AI that can provide a set of narrow tasks where it seems like it is able to think.

In contrast to strong AI, the system can understand well enough and self-improve though its self-thought. Strong is more complex where it can made its on decisive answer based on the AI’s models and thought. Hence, it can answer more complex, cognitive problems where it is difficult to distinguish whether it is process by robots or a human being.

**2. Task environment**

**a) For each of the following activities, give a PAES (Performance, environment, actuators, sensors) description of the task environment and characterize it in term of the proprieties. (2 marks)**

1. **Playing soccer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Agent Type | Performance Measure | Environment | Actuators | Sensors |
| Soccer players | Athletic, strong, endurance, reflects. | Nets, jerseys, soccer ball, soccer field, spectators. | Kicking, running, walking, standing, passing, shooting. | Player’s eyes, touch, ears, sense. |

1. **Playing tennis match**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Agent Type | Performance Measure | Environment | Actuators | Sensors |
| Tennis players | Athletic, strong, endurance, reflects. | Nets, jerseys, soccer ball, tennis field, spectators. | Kicking, running, walking, standing, shooting. | Player’s eyes, touch, ears, sense. |

**3. Informed and Uniformed Search**

1. **Breadth first search. The expansion from the nodes left to right. (6 marks)**
2. **Write down solution.**
3. **Can it find the optimal solution? Explain your answer**

**A close up of a logo

Description automatically generated**

Step 1

* Visited: -
* Queue: A

Step 2

* Visited: A
* Queue: B, F, J

Step 3

* Visited: A, B
* Queue: F, J, C, L

Step 4

* Visited: A, B, F
* Queue: J, C, L, K

Step 5

* Visited: A, B, F, J
* Queue: C, L, K, O

Step 6

* Visited: A, B, F, J, C
* Queue: L, K, O, D, G

Step 7

* Visited: A, B, F, J, C, L
* Queue: K, O, D, G, M

Step 8

* Visited: A, B, F, J, C, L, K
* Queue: O, D, G, M

Step 9

* Visited: A, B, F, J, C, L, K, O
* Queue: D, G, M, R

Step 10

* Visited: A, B, F, J, C, L, K, O, D
* Queue: G, M, R, E, H

Step 11

* Visited: A, B, F, J, C, L, K, O, D, G
* Queue: M, R, E, H

Step 12

* Visited: A, B, F, J, C, L, K, O, D, G, M
* Queue: R, E, H, N, P

Step 13

* Visited: A, B, F, J, C, L, K, O, D, G, M, R
* Queue: E, H, N, P, S

Step 14

* Visited: A. B, F, J, C, L, K, O, D, G, M, R, E
* Queue: H, N, P, S, I

Step 15

* Visited: A. B, F, J, C, L, K, O, D, G, M, R, E, H
* Queue: N, P, S, I

Step 16

* Visited: A. B, F, J, C, L, K, O, D, G, M, R, E, H, N
* Queue: P, S, I

Step 17

* Visited: A, B, F, J, C, L, K, O, D, G, M, R, E, H, N, P
* Queue: S, I, Q

Step 18 (Goal state! Found ‘S’)

* Visited: A, B, F, J, C, L, K, O, D, G, M, R, E, H, N, P, S
* Queue: I, Q

1. **A\* search. (6 marks)**
2. **Write down the solution.**
3. **Can it find the optimal solution? Explain your answer.**

F(n) = G(n) + H(n)

Let g be the cost for the edge

Let h be the cost for the for the vertices

A close up of text on a white background

Description automatically generated

Nodes visited: A, J, B, K, L, C, R, M, G, D, S

Shortest path solution: A, J, O, R, S

**C. Discuss the reason why one search strategy is better than another.**

* 1. **Case 1: Breath First Search vs. A\* Search**
  2. **Case 2: A\* Search with h1 vs. A\* Search with h2.**
  3. **Is h1 is admissible? What happen when it is not admissible? Give an example.**

Breadth first search in comparison to A star is a lot easier to implement in comparison. Breadth first search is an exhaustive way of searching all possible path from the root to its children. While breadth first search can be expensive and takes to search the location. Brute forcing the path can show you all the possible solution that you may want. However, it can take some time to finds its goal due to its time complexity.

A Star allows each node to have assigned weight to be able to know which path the best option is. This in comparison to breadth first search is a lot for optimize when searching because it will search a lot less node because each node and path has its own weight which represent a potential search. Thus, A star will visit a lot less node than the breath first search.

H1 also known as the missing tile solution for A star is not admissible. The solution is not very heuristic in general which can be take time longer to find the path. Which can be similar to breadth first search. Although its much faster than breadth first search it is slower than Manhattan solution of finding the path. For example, the missing tiles weight has a large depth which can cause a slower performance than the Manhattan method. Thus, Manhattan would be a better option for the solution.