

# Introduction to Artificial Intelligence

## Example Quiz 1 - answers (currently being edited!)

---

Justify all answers CONCISELY!

1) a) Number of people on the wrong shore is an admissible heuristic is correct - value is 0 for goal, and we always need to move at least as many people as there are on the wrong shore (note the PATH COST is NUMBER OF PEOPLE moved, not number of trips!)

2) Construct a taxonomic hierarchy for animals, which contains mammals, which contain primates, which in turn contains apes and humans (which are disjoint). Whales are also mammals, which are not primates, and no primate is a whale. All whales can swim. Jesus is a human who can walk on water, but cannot swim. Anything that can swim has a fin. Juju is a whale.

a)

```

animals - mammals - primates - humans (-) Jesus
              \
              whales - apes
                    (-) Juju
  
```

axioms:

```

(forall x) x in mammals => x in animals ; etc.
disjoint({whales, primates}, mammals)
(forall x) x in whales => swims(x)
(forall x) swims(x) => (exists y) fin(y) and part_of(y, x)
  
```

```

fin(x) - true if object is a fin
part_of(x, y) - true if object x is a part of object y
  
```

facts:

```

Jesus in humans.
can_walk_on_water(Jesus) ; predicate signifying that object can walk on water
~swims(Jesus) ; swims - predicate signifying object can swim
Juju in whales
  
```

b) (outlines only - full proof as in answer to exercises)

- 1) Jesus has no fin cannot be proved or disproved
- 2) Jesus is a whale disprove via inheritance and disjointness of whales and primates
- 3) Some animals can swim proved via example of Juju the whale,, thus a mammal, and thus an animal
- 4) Some primates cannot swim proved via example of Jesus who's a human, and thus a primate
- 5) If some primate can walk on water, then there are no primates false - proved by counterexample (Jesus)

3) a) Idea is similar to MINIMAX. In general, for an n-person game (value, best\_value are lists of n elements):

```

best(position, #_of_players, player, depth) {
  if (terminal(position))
    return(value(position));
  best_value[player] = -infinity; best_move = error;
  foreach(move in legal_moves(position, player)) {
    new_position = apply_move(position, move);
    new_value = best(new_position, #_of_players,
                    (player+1) mod #_of_players, depth + 1);
    if (new_value[player] >= best_value[player]) {
  
```

```

        best_value = new_value;
        best_move = move;
    }
}
if(depth == 0)
    do_move(best_move);
else
    return(best_value);
}

```

For b and c, let us name the moves: A's moves are a1 or a2, B's are b1 or b2, and C's c1 or c2.

b) If two players, say B and C, could make deals, this could lead to a different choice of moves and possibly a worse score for A. In addition, A's algorithm would depend on HOW B and C make their deals. For example, in the following, B can agree with C NOT to make move b1 if C agrees NOT to make move c1. In this case BOTH B AND C benefit. This will NOT work without a prior agreement - if each tries direct maximization of individual score, the best they can hope for is 2 each.

0	-	a1	-	b1	-	c1	(5 2 7)
\			\		-	c2	(5 7 2)
\			b2	-	c1	(5 0 7)	
\				-	c2	(0 5 5)	
	a2	-	b1	-	c1	(2 3 3)	
		\		-	c2	(2 3 4)	
		b2	-	c1	(2 0 0)		
			-	c2	(2 0 0)		

As a result of this, A must choose action a2, rather than a1, which would have been optimal without agreements!