Section B (Use a separate answer book for this section)

1 a Fig. 1 shows the starting position of a Markov Decision Process with one agent, denoted ♔, placed at the top-left corner, and one terminal state, denoted 🏁.

The agent can move in three different directions: Right, Up, and Down.

Hitting the wall results in no movement.

The reward associated with the terminal state is $+100$, while the reward associated with each non-terminal state is -1.

♔		
		🏁

Fig. 1: One king

i) Run the value iteration algorithm, under the assumption that the agent has a discounting factor of 1 and goes to the intended direction with probability 0.8 and to each of the other two directions with probability 0.1. Initialise the algorithm specifying a value of 0 for each non-terminal state. Give the value of each state at each stage of the algorithm, stopping when the value of each state is bigger than 0.

ii) Identify the optimal policy, based on the previous calculations.

© Imperial College London 2017

b Fig. 2 shows the starting position of a Markov Decision Process with two agents, denoted ♔ and ♚, respectively.

The agents take turns to move: first ♔, then ♚, then ♔, then ♚, and so forth.

♔ can move in three different directions: Right, Up, and Down.

♚ can only move in one direction: Left.

Hitting the wall results in no movement.

The game ends if one of the two agents, by moving, occupies the square where the other agent is. The agent who occupies the other agent's square first is the winner. The other one is the loser.

If ♔ wins the reward is $+100$. If ♔ loses the reward is -100. The reward associated with each other state is 0.

Rewards for ♚ are the rewards for ♔ times -1 (in words, it is a strictly competitive zero-sum game).

Both for ♔ and ♚ the discounting factor is 1 and the probability to go the intended direction is 1.

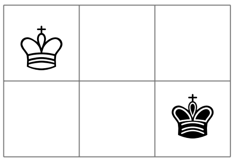

Fig. 2: Two kings

i) Represent the possible states of the game and their transitions.

ii) Identify the optimal policy.

The two parts carry equal marks.