# Reinforcement Learning, Grey-Jack


Last modified on Saturday, 14 May 2022 by f.maire@qut.edu.au

This week we explore RL algorithms through an artificial player for the game *Greyjack* (a simplified version of card game *Blackjack*). You will finish the implementation of two RL methods for a Greyjack artificial player. These RL methods are Monte Carlo and SARSA. You should start with Monte Carlo (the method is simpler and the code provided is almost complete).

- The game Greyjack is played with an infinite deck of cards (i.e. cards are sampled with replacement).
- Each draw from the deck results in a value between 1 and 10 (uniformly distributed) with a colour of red (probability 1/3) or black (probability 2/3).
- There are no aces or picture (face) cards in this game.
- At the start of the game both the player and the dealer draw one black card (fully observed).
- Each turn the player may either ***stick*** or ***hit***.
- If the player hits then she draws another card from the deck.
- If the player sticks she receives no further cards.
- The values of the player's cards are added (black cards) or subtracted (red cards).
- If the player's sum exceeds 21, or becomes less than 1, then she "goes bust" and loses the game (reward -1).
- If the player sticks then the dealer starts taking turns. The dealer always sticks on any sum of 17 or greater, and hits otherwise. If the dealer goes bust, then the player wins; otherwise, the outcome – win (reward +1), lose (reward -1), or draw (reward 0) – is the player with the largest sum.

## Exercise

- Download the provided Python files. The main file is *demo_greyjack.py*, but the file you have to complete is *learning_agent.py.*
- In the file *learning_agent.py* , complete the functions tagged with "INSERT YOUR CODE HERE".