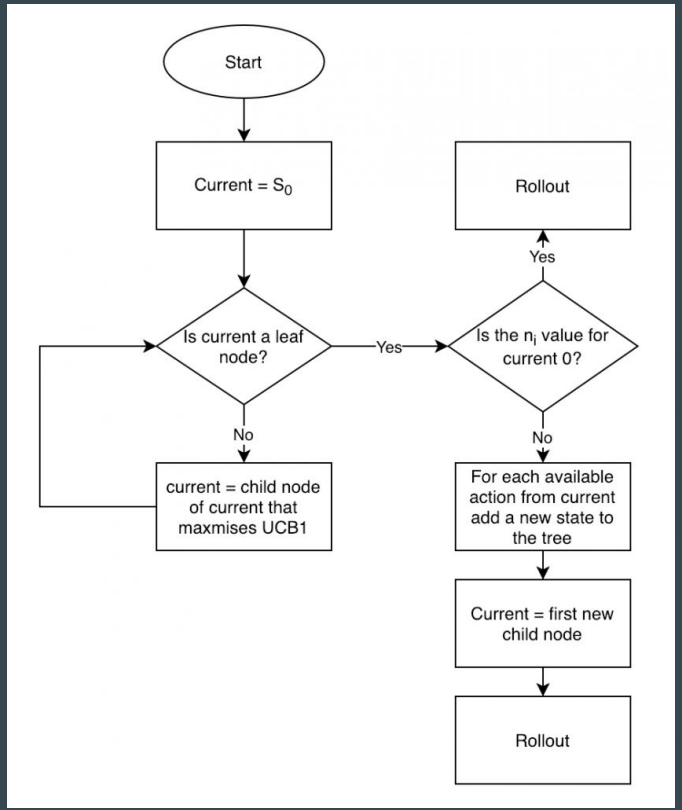
Tablut Challenge 2020

Group *I Serenissimi*
Chinellato Diego and Campardo Giorgia

Python implementation of a MonteCarlo Tree Search

Monte Carlo tree search algorithm:

- **core process**:
  - selecting a leaf
  - expanding it
  - performing a random playout from that leaf
  - backpropagation of the result
- choose the best action after performing as many iterations as it is possible



Python implementation of a MonteCarlo Tree Search

- Implemented variants:
  - best action can be chosen with different policies: *max child, robust child, secure child*
  - when expanding a leaf, if there's a terminal state among its children no playout is executed, instead the value of that state is returned
  - when the tree is built from the current state, if there's a terminal state among its children, the action that leads to that state is chosen
  - parallel playouts
  - checking existence of terminal nodes during random playout
- Other tested variants
  - replacement of the playout with a neural network
  - assign different weights to playouts based on several conditions, e.g. length of the playout, number of simulations