*Introdução*

Descrever os objetivos e motivação do trabalho. Descrever num parágrafo breve a estrutura do resto do relatório.

*Lista de Jogadas Válidas*

Obtenção de uma lista de jogadas possíveis. Exemplo: valid\_moves(+Board, +Player, -ListOfMoves).

Para a validação das jogadas foram usados os seguintes predicados:

checkMove(Board, Player, NewBoard, Expected, ColumnIndex, RowIndex)

isEmptyCell(Board, Row, Column, Res)

isValidPosLines(Board, Row, Column, Res)

isWorkerLines(Board, WorkerRow, WorkerColumn, Row, Column, Res)

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResN, 'N' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResNE, 'NE' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResE, 'E' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResSE, 'SE' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResS, 'S' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResSO, 'SO' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResO, 'O' )

verifyLine(Board, WorkerRow, WorkerColumn, Row, Column, 1, ResNO, 'NO' )

getValueFromMatrix([\_H|T], Row, Column, Value)

**checkMove** – Verifica todas as jogadas tanto do jogador, como do worker. Se não for possível informa o jogador.

**isEmptyCell** – Verifica se a célula (*Row, Column)* está vazia.

**isValidPosLines** – Vai buscar as posições dos workers com a ajuda do predicado *getWorkersPos* e verifica se a célula (*Row, Column*) está na linha de visão de pelo menos um dos workers.

**isWorkerLines** – Verifica se a célula (*Row, Column*) está nalguma das linhas de visão do worker (*WorkerRow, WorkerColumn*) com a ajuda do predicado *verifyLine*.

**verifyLine** – Verifica se a célula está na linha de visão do worker. A direção destas linhas pode ser ‘N’, ‘NE’, ‘S’, ‘SE’, ‘S’, ‘SO’, ‘O’, ‘NO’. Este predicado calcula a linha nessa direção até encontrar uma peça. Foi feito o overload deste predicado para cada uma das direções para o código se tornar mais percetível.

**getValueFromMatrix** – Analisa o que está na célula (*Row, Column*) da matriz. Este predicado não é diretamente validação de jogada mas é bastante relevante para este fim, visto que pode ser usado em duas situações: retorna em *Value* o conteúdo daquela célula, ou, caso *Value* já estiver atribuído a função falha, ajudando o utilizador a saber que o que está na matriz não é o valor esperado.

*Avaliação do Tabuleiro*

Avaliação do estado do jogo, que permitirá comparar a aplicação das diversas jogadas disponíveis. Exemplo: value(+Board, +Player, -Value).

*Jogada do Computador*

Escolha da jogada a efetuar pelo computador, dependendo do nível de dificuldade. Por exemplo: *choose\_move(+Level, +Board, -Move)*.