

Heading

Comp 1 2015 pre-release

Task 2

1. The piece is trying to be moved to a space that is out of range (too large) for the list. You are providing an index which is greater than the horizontal range of the array.
2. The piece is moved to a space off to the left of the board, therefore it is too small for the array and so the index is out of range.
3. The piece is moved to a space moved too far up the board therefore the index is out of range of the vertical part of the array.
4. When you move off the bottom edge of the board the index is too small and out of range for the vertical part of the array.
5. Moving right causes the program to crash with an index error, whereas moving it left causes the program to state if that the move was not legal. When the piece is moved above the board you get a message saying that the move is not valid. Whereas when the piece is moved below the board you get an index error for the rank being out of range.
6. Function responsible = **CheckMoveIsLegal**

Task 3

1. function responsible = **GetMove**

Task 5

1. If a piece is taken the program currently says that the move is invalid and gets you to enter another move.
3. Function for making pieces move = **MakeMove**

Task 6

1. Function responsible for promotion = **MakeMove**

Task 7

1. Function for generating the board = **DisplayBoard**

Task 8

Variable roles

1. Fixed value - A none calculated number that is not changed at all. eg: **BOARDDIMENSION**
2. Stepper - A variable that counts the systematic repetitions of a procedure. eg: **Count**
3. Most recent holder - A variable that holds the most recent number/value given to it from input. eg: **StartSquare**
4. Most wanted variable - This holds the most significant or wanted variable found so far. eg: **Piece**
5. Gatherer - The variable that keeps a running total of different values as they are accumulated. eg: **WhoseTurn**

6. Transformation - A variable that is the result of a calculation from other fixed variables.

eg: **ord(StartSquare)**

7. Follower - A variable that is updated each time the old value is changed from another data item. eg: **FinishSquare**

8. Temporary - The variable that only holds a value for a short time. eg: **WhoseTurn**

Task 9

Functions and parameters

When passing by reference you are passing in where the parameter is stored, so any changes to this parameter in the function are accessible to the function that has called it. However when you pass by value you make a copy of the variable to the parameter so any changes have no effect on the original value.

1. CreateBoard = *Uses neither as no arguments are taken*
2. DisplayWhoseTurnIts = *Value*
3. GetTypeOfGame = *Uses neither as no arguments are taken*
4. DisplayWinner = *Value*
5. CheckIfGameWillBeWon = *Uses both value and reference*
6. DisplayBoard = *Reference*
7. CheckRedumMovelsLegal = *Uses both value and reference*
8. CheckSarrumMovelsLegal = *Uses both value and reference*
9. CheckGisgigirMovelsLegal = *Uses both value and reference*
10. CheckNabuMovelsLegal = *Uses both value and reference*
11. CheckMarzazPaniMovelsLegal = *Uses both value and reference*
12. CheckEtluMovelsLegal = *Uses both value and reference*
13. CheckMovelsLegal = *Uses both value and reference*
14. InitialiseBoard = *Uses both value and reference*
15. GetMove = *Value*
16. ConfirmMove = *Value*
17. GetPieceName = *Value*
18. MakeMove = *Uses both value and reference*

Written with [StackEdit](#).