Task 11

Attempt to explain the purpose of each of the new functions listed above and indicate any parameters or return values that each of them will require.

display_menu(): Displays a menu in a numbered format on screen. Takes in no parameter and returns no values.

get_menu_selection(): Asks the user for the selection from the list that has been displayed and assigns the input to a variable. It takes in no parameters but returns the variable selection.

make_selection(): Carries out the correct instruction relative to the selection. Takes in selection as a parameter but returns no values.

play_game(): Runs the whole game using multiple different functions.Takes in selection as a parameter but returns no values.

Task 12

Identify the functions that will require modification to make it possible for an in-game menu to be presented.

GetMove(StartSquare, FinishSquare, WhoseTurn)

Task 13 Surrendering

Identify the functions that will require modification to make it possible to surrender during the game. Explain why each function will require modification.

get_pause_menu_selection(WhoseTurn), **surrender(WhoseTurn)** and **GetMove(StartSquare, FinishSquare, WhoseTurn)** are the function that needs to be modified. get_pause_menu_selection(WhoseTurn) needa to change as it need to take in WhoseTurn as a parameter for it to work with the surrender(WhoseTurn) function. The surrender(WhoseTurn) is a brand new function that takes in WhoseTurn to determine who is surrendering and who wins from it. Finally GetMove(StartSquare, FinishSquare, WhoseTurn) needs to change slightly as the get_pause_menu_selection() needs to taken in WhoseTurn so it will pass it from play_game() to get_move() to get_pause_menu_selection() then finally carry out the surrender if necessary.

Task 14 Refactoring

Explain what is meant by the term refactoring and why it is sometimes useful to refactor sections of code.

Refactoring means reconstructing code without changing the behaviour of it. It is useful as you can split code in to useful part as you may only need to use part of a function.

Task 15

Describe each variable role in your own word

Fixed Value: A varible that holds a value that is either hardcoded or inputed by the user but with no calculations or changes

Stepper: A variable that acts as a counter when iterating through a loop giving a systamatic value each loop

Most recent holder: A variable that stores the most recent value at any point when processing multiple values to assign to a variable

Most wanted holder: A variable that holds the correct value(s) from a list of variables when we given a condition. E.g smallest number

Gather: A variable that holds a value which is accumulated after effects of other values

Transformation: A variable that changes after a stated calculation with other variable

Follower: A variable that is updated with the leftover value of another data item

Temporary: A variable that hold a value for a limited time

Give an example of variable from the program code for each variable role.

Fixed Value: BOARDDIMENSION = 8

Stepper: for Count in range(BOARDDIMENSION + 1):

Most recent holder: StartRank = StartSquare % 10

Most wanted holder:

Gather:

Transformation: FinishRank = FinishSquare % 10

Follower:

Temporary: StartSquare = int(input("Enter coordinates of square containing piece to move (file first): "))

Task 20

Describe the difference between passing by value and passing by

reference in your own words. Passing by reference means passing in a variable name that is assigned to a value to use. We can then return this same variable and the changes made to it will be updated. Passing by value is passing in an arguement to a function which creates a copy for the function to process. We can return the outcome of the function but the original argument will stay the same.

For each function in the program identify the mechanism using to pass each parameter.

CreateBoard() = None

DisplayWhoseTurnItIs(WhoseTurn) = By Value

GetTypeOfGame() = None

DisplayWinner(WhoseTurn) = By Value

CheckIfGameWillBeWon(Board, FinishRank, FinishFile) = By Value and Reference

DisplayBoard(Board) = By Reference

CheckRedumMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile, ColourOfPiece) = By Value and Reference

CheckSarrumMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

CheckGisgigirMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

CheckNabuMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

CheckMarzazPaniMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

CheckEtluMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

CheckMoveIsLegal(Board, StartRank, StartFile, FinishRank, FinishFile, WhoseTurn) = By Value and Reference

InitialiseBoard(Board) = By Reference

InitialiseNewBoard(Board) = By Reference

InitialiseSampleBoard(Board) = By Reference

GetMove(StartSquare, FinishSquare) = By Value

MakeMove(Board, StartRank, StartFile, FinishRank, FinishFile,

WhoseTurn) = By Value and Reference

ConfirmMove(StartSquare, FinishSquare) = By Value

ConfirmMove(StartSquare, FinishSquare) = By Value

GetPieceName(Board, StartRank, StartFile, FinishRank, FinishFile) = By Value and Reference

display_menu() = None

get_menu_selection() = None

display_pause_menu() = None

get_pause_menu_selection(WhoseTurn) = By Value

surrender(WhoseTurn) = By Value

play_game(selection) = By Value

make_selection(selection) = By Value

main_menu() = None