

Brute Force Algorithm

Create a PossibleWords list

Check for items in the Dictionary that

Can be made from the tiles in myTiles and on the Board and Can fit on the board

If these conditions are met then append it to the PossibleWords list

if Move is 1 then:

set Initial Max Score to 0

set Initial Max Item to ''

Check for words in PossibleWords

If the score of the word is greater than the current Max Score value and the length of the word is smaller than or equal to the Board Size less half of the Board Size then

Set the Max Score to the score of the current word

Set the Max Item to the string of the current word

If the score of the Userword is equal to Max Score then

Print 'Your move was the best move. Well done!'

Else then

Print(The Maximum Scoring word is (MaxItem) of score (MaxScore) at position (Half of the BoardSize): (Half of the BoardSize) : Direction of choice

If Move is greater than 1 then:

Until the MaxWord has been found then:

Set Initial Max Score to 0

Set Initial Max Item to ''

Check for words in PossibleWords

Define NeededTiles as a list of letters that are used to make the word being check and from myTiles

If the score of the word is greater than the current Max Score value and the length of the word is smaller than or equal to the Board Size less half of the Board Size then

Set the Max Score to the score of the current word

Set the Max Item to the string of the current word

To check if the word can be made horizontally or vertically;

Check every Column in between 0 to Boardsize less the length of the word(inclusive) against

Checking every Row in between 0 to Boardsize

Define TilesReplaced as 0

Check each letter in the word against each Row and Column.

If the Board is empty in that position;

Add 1 to TilesReplaced

The word is connected in a valid location

Otherwise if the Board has a character in it and it isn't a shareable letter that the word can use then

The word is not connected in a valid location

If the Number of Tiles replaced is equal to the length of the word than the user has entered the word in an empty location and not connected to a tile therefore

The word is not connected to a tile

If the Tile is empty than

if the Board slot is not empty, check if the slot has a letter than is the same index in the UserWord as it does from the number of slots it is away from Col1 or Rows (depending on direction)

If not than:

The word is immediately not in a valid location

Define a list called MaxNeededTiles which will contain all the letters that is required from myTiles to make the word

If the Tile being assessed is empty than

Append the letters in the MaxItem into a list called MaxNeededTiles

If MaxNeededTiles is not in myTiles than

The word cannot be made from myTiles

Otherwise if MaxNeededTiles can be made from myTiles than

The word can be made from myTiles

If All the above conditions are met than:

Print('The Max Word Possible is',(MaxItem),'of a score of',(MaxScore),'at position', (position where Row and Column where valid) , (at direction H or V depending on which direction is being assessed)
Set FoundMaxWord to True to end the While loop

If not than the Word is not a valid solution:

Remove the word from the PossibleWords list and let the code loop into checking the next highest scoring PossibleWord

Justification:

My Strategy finds the Maximum Scoring solution because it checks through every possible word that can be made using words on the board and in myTiles, than it checks through all of these words in a descending order(so starting with the highest scoring word) than checks if it can meet through all of the conditions required for it to be able to be placed in a Valid Location. It checks through each possible Row and Column such that the word can be fit either horizontally or vertically (If both than my code will print the horizontal solution only). If the word cannot be placed in a Valid Location on the Board than it removes the word from the PossibleWords list and checks the next highest word instead. It repeats this process until it finds a word and a location where it is valid, than the loop ends and the User is presented with the Maximum Scoring word, its score and the location where it is valid.