**Coding challenge**

Challenge - 1 Should you capture?

Problem Statement:

Assumptions:

Constraints:

**Topics that this problem covers:** Solution:

**Challenge - 1: Should you capture?**

**Problem Statement:**

A typical game of Chess consists of 6 pieces namely: King, Queen, Rook, Knight, Bishop and Pawn. The game ends when the opposite side White or Black) is able to capture the King making it an invaluable piece but the rest of the pieces have a certain point value associated with them which is shown below:

Queen - 9 pts

Rook - 5 pts

Knight - 3 pts

Bishop - 3 pts

Pawn - 1 pts

The objective of this challenge is to figure out whether to consider a particular trade of pieces or not.

You have an input which contains the opportunity that a number of White's pieces have against a number of Black's pieces. Your task is to compute whether a particular trade is favorable to White or Black and by how many points.

Sample Input: 1 K 4 P

Sample Output: Trade is favourable to White by 1 pt.

*Explanation*: The input given in order is : White: 1 Knight against Black: 4 Pawns.

If white's Knight captures 4 pawns of black before getting captured itself, White gains 4 points. White also loses 3 points because the Knight gets captured in the process making the trade favorable to White by 43 1 pt.

Sample Input: 2 R 1 Q

Sample Output: Trade is favourable to Black by 1 pt

*Explanation*: The input given in order is - White: 2 Rooks against Black: 1 Queen.

If the Rooks of White together capture the black Queen before getting captured themselves , White gain 9 pts. White also loses 5510 pts because the rooks get captured in the process making the trade favorable to White by 109 = 1 pts which is equivalent to making the trade favorable to Black by 1 pts.

**Assumptions:**

 Seemingly impossible trades like 5 P 8 P are possible.

 The trade advantage is purely numerical and does not take into account positional considerations.

**Constraints:**

 Pieces = Queen(Q), Rook(R), Knight(K), Bishop(B), Pawn(P)}

 Number of pieces per side per type ≤ 8

***Note: If the value turns out to be the same for both side captures, then print: "The trade is equally favorable to both Black and White."* Topics that this problem covers:**

 **if-else-elif**

 **Operators**

 **Strings**

 **Data Structures**

**Solution:**

**Approach - 1** **Naïve Approach**

Most beginners are going to create a boat load of if-else statements which would solve the problem but isn't optimal in any sense. Instead what we will do is take advantage of the inbuilt data structures that Python offers us.

**Approach - 2** **Using Hash-map Dictionary)**

nw,white,nb,black = input("Enter the pieces that could be traded. White followed by Black.").split()

point = {'P': 1,

'K': 3,

'B': 3,

'R': 5,

'Q': 9

}

white\_gain = point[black]\*int(nb)

black\_gain = point[white]\*int(nw)

if white\_gain==black\_gain:

print("The trade is equally favourable to both Black and White. ")

elif white\_gain<black\_gain:

print(f"The trade is favourable to Black by {black\_gain - white\_gain} points.")

else:

print(f"The trade is favourable to White by {white\_gain - black\_gain} points.")