Algorithm

Step 1 : Start

Step 2 : create cards(6)

Step 3 : declare empty list outdated, and round_count, scores,special_spells, prev_round_winner

Step 4 : Shuffle the cards

shuffle(deck)

Step 5 : Roll a dice using

random.randint(1,6)

Step 5.1 : Display the results

Step 6 : Append the outdated deck and reshuffle

self.outdated.append(char)

shuffle(self.outdated)

Step 7 : randomly pick a card and update outdated_deck

Step 8 : randomly select an integer r and

if r is 1 then "god spell"

if r is 2 then "resurrect spell"

else return None

Step 9 : if spell is resurrect then pick one random card from outdated cards and players card

if spell is god then pick a random card from players cardsAlgorithm

Step 1 : Start

Step 2 : create cards(6)

Step 3 : declare empty list outdated, and round_count, scores,special_spells, prev_round_winner

Step 4 : Shuffle the cards

shuffle(deck)

Step 5 : Roll a dice using

random.randint(1,6)

Step 5.1 : Display the results

Step 6 : Append the outdated deck and reshuffle

self.outdated.append(char)

shuffle(self.outdated)

Step 7 : randomly pick a card and update outdated_deck

Step 8 : randomly select an integer r and

if r is 1 then "god spell"

if r is 2 then "resurrect spell"

else return None

Step 9 : if spell is resurrect then pick one random card from outdated cards and players card

if spell is god then pick a random card from players cards

Step 10 : compare the characteristics and decide the winner update round_count

Step 11 : Display the Winner(player1 / player 2/ tie)

Step 10 : compare the characteristics and decide the winner update round_count

Step 11 : Display the Winner(player1 / player 2/ tie)