10-20-30

Team 26

Tejaswini : 20WH1A0472 : ECE
Manvitha : 20WH1A66 55: CSM
Sony :20WH1A0201 :EEE
Jhansi : 20WH1A0540 : CSE
Jeevitha : 20WH1A1233 : IT

BVRIT HYDERABAD College of Engineering for Women.

October 25, 2021

Problem Statement

- A simple solitaire card game called 10-20-30 uses a deck of 52 cards in which suit is irrelavant.For each card placed on a pile check that the pile to see combination total is 10,20 or 30. Eliminate the deck or else add one card to each pile until the cards in the deck are over.You win if all the piles are eliminated.You loose if your anable to eliminate a card and we have a posibility to have a draw if niether of the previous 2 conditions ever occurs.

Rules

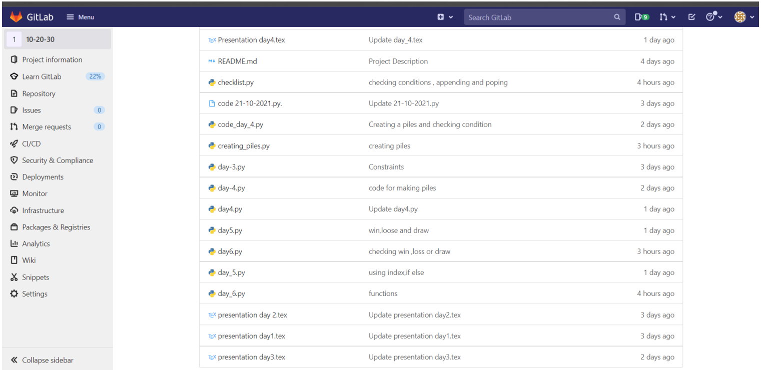
- The value of a face card (king,queen,jack) is 10 and ace is one.
- Begin by dealing out seven cards, left to right forming seven piles. After playing a card on the rightmost pile, the next pile upon which you play a card is the leftmost pile.

- For each card placed on a pile, check that the pile to see if one of the following three cards combinations totals 10 ,20 or 30.
    - The first two cards and last one card.
    - The first one card and last two cards.
    - The last three cards.
- If so , pickup the three cards and place them on the bottom of the deck.

APPROACH

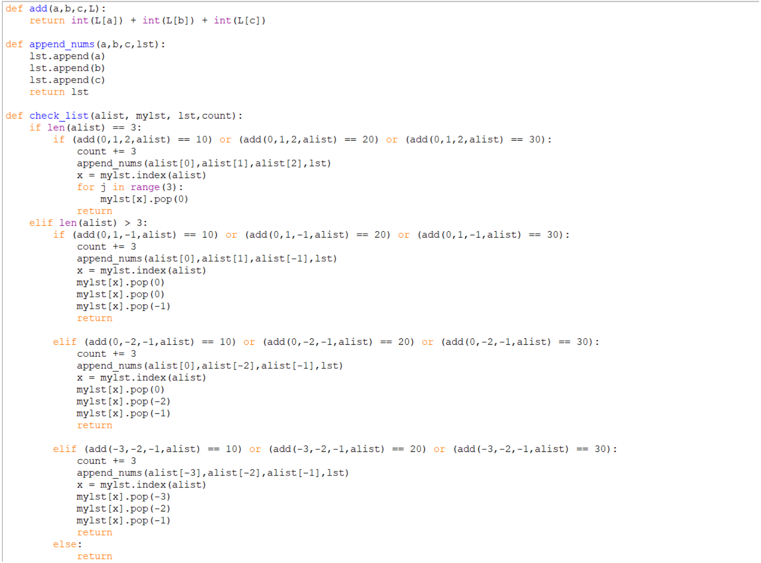
- TASK-1 We intend to approach this problem statement by using basic understanding of data types
- TASK-2 We approached the logic of the problem through manual work

GIT Repo



Presentation day4.tex	Update day_4.tex	1 day ago
README.md	Project Description	4 days ago
checklist.py	checking conditions , appending and poping	4 hours ago
code 21-10-2021.py.	Update 21-10-2021.py	3 days ago
code_day_4.py	Creating a piles and checking condition	2 days ago
creating_piles.py	creating piles	3 hours ago
day-3.py	Constraints	3 days ago
day-4.py	code for making piles	2 days ago
day4.py	Update day4.py	1 day ago
day5.py	win,loose and draw	1 day ago
day6.py	checking win ,loss or draw	3 hours ago
day_5.py	using index,if else	1 day ago
day_6.py	functions	4 hours ago
presentation day 2.tex	Update presentation day2.tex	3 days ago
presentation day1.tex	Update presentation day1.tex	3 days ago
presentation day3.tex	Update presentation day3.tex	2 days ago

code



```
def add(a,b,c,L):
    return int(L[a]) + int(L[b]) + int(L[c])

def append_nums(a,b,c,lst):
    lst.append(a)
    lst.append(b)
    lst.append(c)
    return lst

def check_list(alist, mylst, lst,count):
    if len(alist) == 3:
        if (add(0,1,2,alist) == 10) or (add(0,1,2,alist) == 20) or (add(0,1,2,alist) == 30):
            count += 3
            append_nums(alist[0],alist[1],alist[2],lst)
            x = mylst.index(alist)
            for j in range(3):
                mylst[x].pop(0)
            return
    elif len(alist) > 3:
        if (add(0,1,-1,alist) == 10) or (add(0,1,-1,alist) == 20) or (add(0,1,-1,alist) == 30):
            count += 3
            append_nums(alist[0],alist[1],alist[-1],lst)
            x = mylst.index(alist)
            mylst[x].pop(0)
            mylst[x].pop(0)
            mylst[x].pop(-1)
            return

        elif (add(0,-2,-1,alist) == 10) or (add(0,-2,-1,alist) == 20) or (add(0,-2,-1,alist) == 30):
            count += 3
            append_nums(alist[0],alist[-2],alist[-1],lst)
            x = mylst.index(alist)
            mylst[x].pop(0)
            mylst[x].pop(-2)
            mylst[x].pop(-1)
            return

        elif (add(-3,-2,-1,alist) == 10) or (add(-3,-2,-1,alist) == 20) or (add(-3,-2,-1,alist) == 30):
            count += 3
            append_nums(alist[-3],alist[-2],alist[-1],lst)
            x = mylst.index(alist)
            mylst[x].pop(-3)
            mylst[x].pop(-2)
            mylst[x].pop(-1)
            return
        else:
            return
        elif (add(0,-2,-1,alist) == 10) or (add(0,-2,-1,alist) == 20) or (add(0,-2,-1,alist) == 30):
            count += 3
            append_nums(alist[0],alist[-2],alist[-1],lst)
            x = mylst.index(alist)
            mylst[x].pop(0)
            mylst[x].pop(-2)
            mylst[x].pop(-1)
            return

        elif (add(-3,-2,-1,alist) == 10) or (add(-3,-2,-1,alist) == 20) or (add(-3,-2,-1,alist) == 30):
            count += 3
            append_nums(alist[-3],alist[-2],alist[-1],lst)
            x = mylst.index(alist)
            mylst[x].pop(-3)
            mylst[x].pop(-2)
            mylst[x].pop(-1)
            return
        else:
            return
    else:
        return

lst = list(map(int, input("Enter 52 values: ").split()))
mylst = [[] for i in range(7)]
count = 0

index = 0
while True:
    if lst == []:
        t = index
        break
    ele = lst.pop(0)
    i = index%7
    index += 1
    mylst[i].append(ele)
    check_list(mylst[i], mylst, lst,count)
    if mylst[i] == []:
        mylst[i].clear
if lst == []:
    if t == count:
        print("Win:",t)
    elif t % 7 == 6:
        print("Loss:",t)
    else:
        print("Draw:",t)
```

Learnings

- Pushing the files into Git Repo
- Making Presentation using Latex.

Challenges

- Installing all the required Packages in windows.
- Pushing the Folders into git Repo.

Statistics

- Number of Lines of Code : 76
- Number of Functions : 3

Contribution

- 20wh1a1233-Logic and Presentation
- 20wh1a0201-Logic and Writing of the code
- 20wh1a6655-Logic and Writing of the code
- 20wh1a0540-Writing of the code and manual work
- 20wh1a0472-LaTeX and logic of the code

THANK YOU