from sudoku import *

def play_game(filename):
    
    Begins user interaction with Sudoku game

    Single input of filename of particular sudoku game data

    global status #assigning global and local variables
    status=0
    failed=0
    file_name=str(filename)+'.htm' #automatically adds file extension
    game=read_game(file_name)
    generate_grid(game)

    while status == 0: #beginning of game code, if status == 1 end game()
        try:
            if status == 1: #asking if game is complete
                break #ending process as game has finished
            input=list(input()) #asking for command from user
            command=list(input())
            if command[0] == 'a': #processing command as 'add value'
                if command[1] == ' ': # Starts procedure a (add value)
                    r=command[2]
                    c=command[4]
                    ch=command[6]
                    inputcheck(r,c,ch) #checks if user input is valid
                    addvalue(int(r), int(c), int(ch), game)
                    generate_grid(game) #generates output for user interaction
                    command[0]=''
                    if command[0] == 'f': #processing command as 'autofill'
                        autofill(game)
            command[0]=''
            if command[0] == ' ': #if command is empty - game continues
                continue
        except:
            continue

    win=0 # Game has not been Won
    nodeal=0 # Used to check for Errors in Input
    realname=str(filename)+'.htm' # Adds .htm to input string
    game=read_game(realname) # Sets the game list
    generate_grid(game)
    while win == 0:
        try:
            if win == 1: #If the game has been won, end loop
                break
            input=raw_input('Command: ')
            x=list(input()) #Converts input to a List
            if x[0] == 'a':

                if x[1] == ' ': # Starts procedure a (add value)
                    r=x[2]
                    c=x[4]
                    choi=x[6]
                    checknums(r,c,choi)
                    addvalue(int(x[2]), int(x[4]), int(x[6]), game)
                    generate_grid(game)
                    x[0]=''

            elif x[1] == 'f': # Starts procedure af (Auto-fill)
                autofill(game)

                x[0]=''
                if x[0] == ' ': #if command is empty - game continues
                    continue
                else:
                    nodeal=1