Flames Game Python Code :

```
x=input("Enter Boy Name:")
y=input("Enter Girl Name:")
x=x.replace(" ","")
y=y.replace(" ","")
x=list(x)
y=list(y)
for i in x[:]:
    if i in y:
        x.remove(i)
        y.remove(i)
count=len(x)+len(y)
answer = ["Friends", "Love", "Affection", "Marriage", "Enemy",
"Siblings"]
while len(answer) > 1 :
        split_index = (count % len(answer) - 1)
        if (split_index>=0) :
                right = answer[split_index + 1 : ]
                left = answer[ : split_index]
                answer = right + left
        else :
                answer = answer[ : len(answer) - 1]
print("Reltionship Status : ",answer)
```

Hangman Game:

```
import random
def greetings():
        username = input(" Enter your name:    ").capitalize()
        if username.isalpha() == True:
                print("Hello",username,",All the Best !")


        else:
                print('Please enter a valid name using alphabets only')
                username = input('Enter a game name here:      ')
                print('Hello, ' + username + '! Please go through the rules of
the game below:')
def getWord():
        listOfWords = ['Apple', 'Papaya', 'Mango', 'Kiwi', 'Orange',
'Pineapple', 'Grapes', 'Cherry', 'Watermelon']
        return random.choice(listOfWords).lower()
def gameRun():
        greetings()
        alphabet = ('abcdefghijklmnopqrstuvwxyz')
        randomWord = getWord()
        guessedLetters = []
        attempts = 6
        guess = False
        print()
        print('The Word consists of', len(randomWord), 'letters.')
        print(len(randomWord) * '_')

        while guess == False and attempts > 0:
                print('You have ' + str(attempts) + ' attempts')
                userGuess = input('Guess a letter in the word or enter the full
word: ').lower()
```

```
        if len(userGuess)  == 1:
            if userGuess  in guessedLetters:
                print('You  have already guessed  that letter  before.
Try again!')
            elif userGuess  not in randomWord:
                print('Oops!  This letter  is not a part of a word.')
                guessedLetters.append(userGuess)
                attempts  -= 1
            elif userGuess  in randomWord:
                print('This  letter is present  in the word!')
                guessedLetters.append(userGuess)
            else:
                print('Invalid  Input!  You  might have  entered  the
wrong entry.')
        elif len(userGuess)  == len(randomWord):
            if userGuess == randomWord:
                print('Great,  Right Guess  !!!')
                guess = True
            else:
                print('Oops!  Wrong Guess')
```

```
                    attempts -= 1
            else:
                print('The length of the guess is not the same as the
length of the word.')
                attempts -= 1


            the_status = ''
            if guess == False:
                for letter in randomWord:
                    if letter in guessedLetters:
                        the_status += letter
                    else:
                        the_status += '_'
                print(the_status)


            if the_status == randomWord:
                print('Right Guess !!! Congratulations :)')
                guess = True
            elif attempts == 0:
                print('Oh my Bad , You ran out of Guesses')
```

```
gameRun()
```

Magic Square:

```
def generateSquare(n):
    magicSquare = [[0 for x in range(n)]
                            for y in range(n)]
    i = n / 2
    j = n - 1
    num = 1
    while num <= (n * n):
        if i == -1 and j == n:
            j = n - 2
            i = 0
        else:
            if j == n:
                j = 0
            if i < 0:
            i = n - 1
        if magicSquare[int(i)][int(j)]:
            j = j - 2
            i = i + 1
            continue
        else:
            magicSquare[int(i)][int(j)]  = num
            num = num + 1
        j = j + 1
        i = i - 1
    print ("Magic Square for n =", n)
    print ("Sum of each row or column",n * (n * n + 1) / 2, "\n")
    for i in range(0, n):
        for j in range(0, n):
            print('%2d ' % (magicSquare[i][j]),end  = '')
            if j == n - 1:
                print()
n=int(input("Number  of rows of the Magic Square:"))
generateSquare(n)
```

Snake and Ladder Game :

```
class QueueEntry(object):
    def __init__(self, v=0, dist=0):
        self.v = v
        self.dist = dist
def getMinDiceThrows(move, N):
    visited = [False] * N
    queue = []
    visited[0] = True
    queue.append(QueueEntry(0, 0))
    qe = QueueEntry()
    while queue:
        qe = queue.pop(0)
        v = qe.v
        if v == N - 1:
            break
        j = v + 1
```

```
            while j <= v + 6 and j < N:
                if visited[j] is False:
                    a = QueueEntry()
                    a.dist = qe.dist + 1
                    visited[j] = True
                    a.v = move[j] if move[j] != -1 else j
                    queue.append(a)
                j += 1
        return qe.dist
N = 30
moves = [-1] * N
moves[11] = 22
moves[3] = 8
moves[5] = 26
moves[20] = 29
moves[17] = 4
moves[19] = 7
moves[21] = 9
moves[27] = 1
print("Min Dice throws required is {0}".
format(getMinDiceThrows(moves, N)))
```