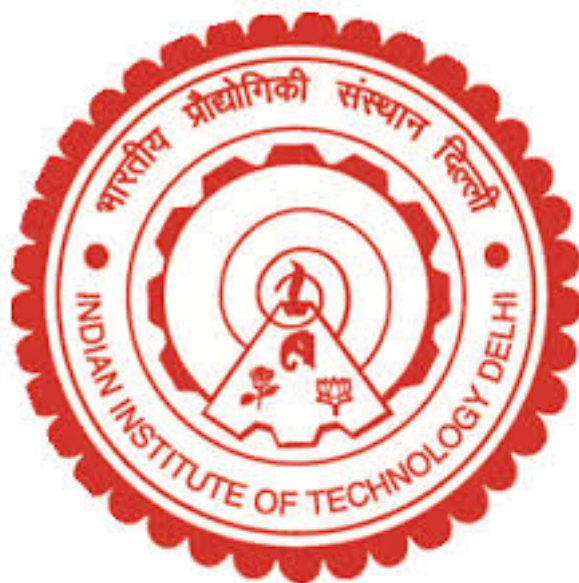


## Assignment-8

**ELP - 718 Telecomm Software Laboratory**

**Manas Kankane  
2019JTM2166  
2019-2021**

A report presented for the assignment on  
Python



**Bharti School Of  
Telecommunication Technology and Management  
IIT Delhi  
India  
Sep 18, 2019**

# Contents

<b>1 Problem Statement-1</b>	<b>5</b>
1.1 Problem Statement . . . . .	5
1.2 Assumptions . . . . .	5
1.3 Algorithm and Implementation . . . . .	5
1.4 Program Structure of PS1 . . . . .	6
1.5 Difficulties Issued . . . . .	7
1.6 Input-Output Format . . . . .	7
1.6.1 Input Format . . . . .	7
1.6.2 Output Format . . . . .	7
1.6.3 Sample Input . . . . .	7
1.6.4 Sample Output . . . . .	7
1.7 Screenshot of PS1 . . . . .	7
<b>2 Problem Statement-2</b>	<b>8</b>
2.1 Problem Statement . . . . .	8
2.2 Assumptions . . . . .	8
2.3 Algorithm and Implementation . . . . .	8
2.4 Difficulties Issued . . . . .	9
2.5 Input-Output Format . . . . .	9
2.5.1 Input Format . . . . .	9
2.5.2 Output Format . . . . .	9
2.6 Program Structure . . . . .	10
2.7 Screenshot of PS2 . . . . .	11
<b>3 GIT Screenshots</b>	<b>11</b>
<b>4 Appendix</b>	<b>13</b>
4.1 Appendix-ps1 . . . . .	13

4.2	<a href="#">Appendix-ps2</a> . . . . .	13
-----	--	----

List of Figures

1	Flow Chart ps1 . . . . .	6
2	Screenshot of PS1 . . . . .	7
3	Screenshot of PS2 . . . . .	11
4	Screenshot of git1 . . . . .	11
5	Screenshot of git2 . . . . .	12
6	Screenshot of git3 . . . . .	12

# 1 Problem Statement-1

## 1.1 Problem Statement

**Parity Check** The simplest way of error detection is to append a single bit, called a parity check, to a string of data bits. This parity check bit has the value **1** if the number of 1's in the bit string is even and has the value **0** otherwise, i.e., Odd Parity Check.

**Bit-Oriented Framing** Data Link Layer needs to pack bits into frames so that each frame is distinguishable from another. Frames can be fixed or variable size. In variable size framing, we define the end of the frame using a bit-oriented approach. It uses a special string of bits, called a flag for both idle fills and to indicate the beginning and the ending of frames.

The bit stuffing rule is to insert a **0** after each appearance of **010** in the original data.

The string **0101** is used as the bit string or flag to indicate the end of the frame.

## 1.2 Assumptions

- We are assuming that if number of **one's** is **even** then we add '**1**'.
- We are assuming that if number of **one's** is **odd** then we add '**0**'
- If we get '**010**' in the bit we have to append '**0**' after that '**010**' bit.

## 1.3 Algorithm and Implementation

1. Enter the parity bit and we save it as a string.
2. Then we declare variables and lists.
3. Then we use **for loop** for inserting bits one by one and then checking it if '1' or '0' using **if-else**.
4. Then be also add variable inside **if i=='1'** and then check it for even and odd by dividing it by 2 using **if-else command**.
5. Then we append '1' or '0' inside if-else as per condition.
6. Then we print the parity.
7. For 2nd part we just append one by one bit into list using for and **append(i)**.
8. Then we check the list one by one using 3 **if-else command** and inserting '1' using **.insert(...)** command and also saving values where we appending '1' in **list y**.
9. Then we put those **y values** and change the '**0**' value.
10. Then we append '0101' in the end
11. Then we print the Transmitting data.

## 1.4 Program Structure of PS1

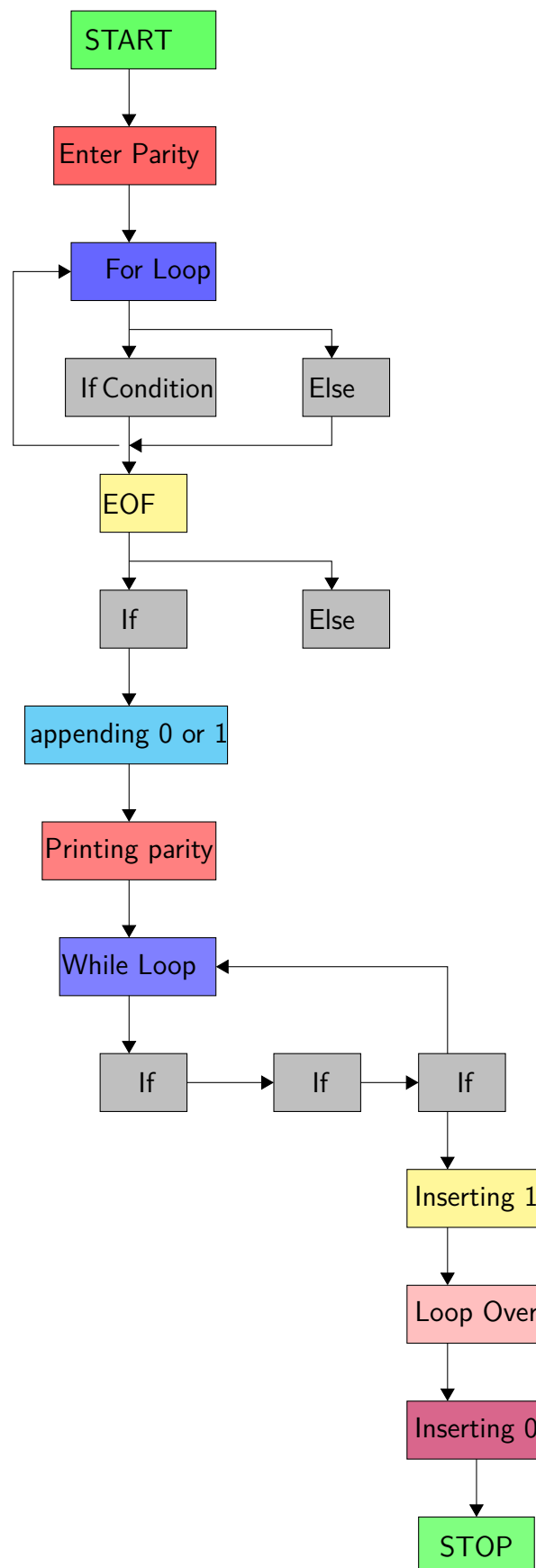


Figure 1: Flow Chart ps1

## 1.5 Difficulties Issued

- For checking even-odd number of 1's.
- For checking '010' in the parity bits.
- For appending into a string from list.

## 1.6 Input-Output Format

### 1.6.1 Input Format

Enter binary bit data that has to be transmitted.

### 1.6.2 Output Format

Print binary bit data with parity bit.

Print the modified string that is to be transmitted

### 1.6.3 Sample Input

010101110100101

### 1.6.4 Sample Output

Parity bit data :010101110100101**1**

Transmitting data:010**0**10111010**0**010**0**11**0**1**0**1

## 1.7 Screenshot of PS1

```
/home/manaskankane/2019JTM2166_Assignment-8/v/bin/python /home/manaskankane/2019JTM2166_Assignment-8/v/bin/python
Enter the parity bit data : 010101110100101

Parity bit data is 0101011101001011

Transmitting data : 01001011101000100110101
```

Figure 2: Screenshot of PS1

## 2 Problem Statement-2

### 2.1 Problem Statement

**3X3 Numeric Tic-Tac-Toe** (Use numbers 1 to 9 instead of X's and O's) One player plays with the odd numbers (1, 3, 5, 7, 9) and the other player plays with the even numbers (2,4,6,8). All numbers can be used only once. The player who puts down 15 points in a line wins (sum of 3 numbers). Always Player with odd numbers starts the game. Once a line contains two numbers whose sum is 15 or greater, there is no way to complete that line, although filling in the remaining cells might be necessary to complete a different line.

Note – Line can be horizontal, vertical or diagonal

#### Constraints:

- $1 \leq \text{Position} \leq 9$
- $1 \leq \text{Number} \leq 9$

#### Terminal:

- Print 'Welcome to the Game!'
- Print whether it is Player 1's or Player 2's chance.
- Get the position and number to be entered from the user.
- Show tic tac toe with data
- Continue till the game gets draw or some player wins and show the result.
- Ask the user whether to continue for the next game or exit.

### 2.2 Assumptions

- We have given a position not less than or equal 1 and not greater than equal to 9.
- We have also given the numbers should be between 1 to 9 with 1 and 9 included.

### 2.3 Algorithm and Implementation

1. We first declare the list and variable.
2. Then we use **while 1**.
3. Then we ask for choice, players name.
4. Then we slice the list and save it into a string and printing it.
5. Then we use while loop and ask for position and number of player1.



6. Then we insert it into a list and convert a list into a string.
7. Then we add the list numbers and check it for 15, if then player1 wins.
8. If all the rows and coloums sum is greater than 15 then match is draw.
9. Same for player2 and we have taken if-else for checking the position.
10. We have also added the constraint for position, match draw and for exit also.

## 2.4 Difficulties Issued

- For slicing the list into string.
- For taking while loops and closing it.
- Sending the data from list to string and printing it.
- Adding all the numbers we put and checking for 15 or greater than 15.

## 2.5 Input-Output Format

### 2.5.1 Input Format

Enter your choice:

Enter the position and number:

Like this upto 9 turns for both player1 and player2

### 2.5.2 Output Format

Enter your choice: 1

Enter the position and number: 5,3

000

030

000

Enter the position and number: 7,4

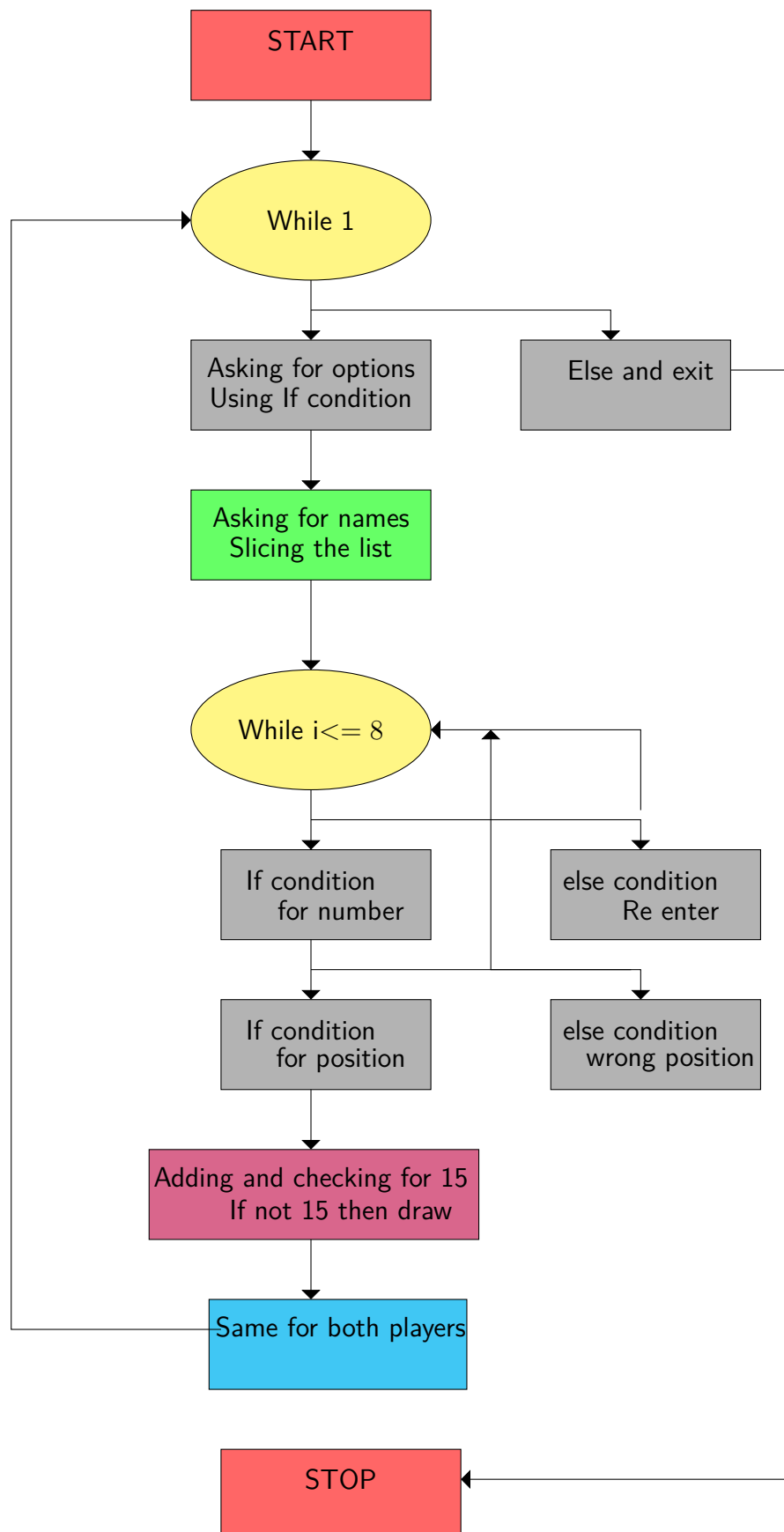
000

030

400

and soon upto 9 turns.

## 2.6 Program Structure



## 2.7 Screenshot of PS2

```

/home/manaskankane/2019JTM2166_Assignment-8/v/bin/python /home/manaskankane/2019JTM2166_Assignment-8/Assignment_8
-----
Welcome to the Game
-----

1. Play Tic-Tac-Toe Game
2. Exit

Enter your choice : 1

Enter player 1 name : manas

Enter player 2 name : anjay

What you want to become even player or odd player : odd

Player manas is odd player and Player anjay is even player

-----Game Started-----

000
000
000
Enter position player manas : 1
Enter number player manas : 5
500
000
000
Enter position player anjay : 2
Enter number player anjay : 5
550
000
000
Enter position player manas : 3
Enter number player manas : 5
555
000
000
Player manas Wins

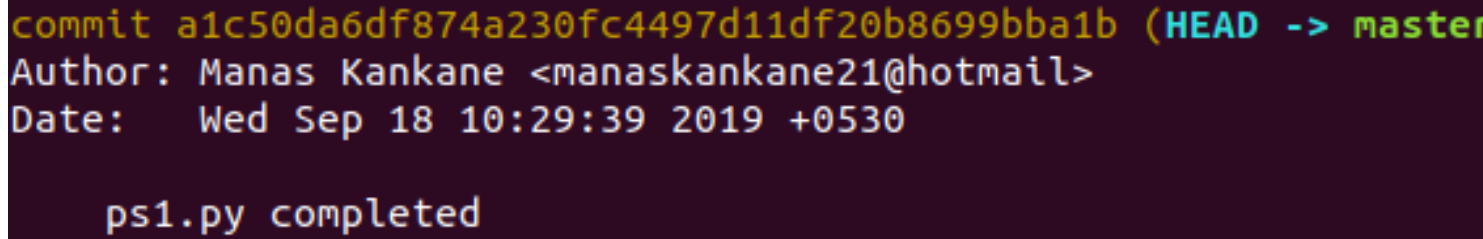
1. Play Tic-Tac-Toe Game
2. Exit

Enter your choice : 2
-----
Thank You for Playing Game
-----

```

Figure 3: Screenshot of PS2

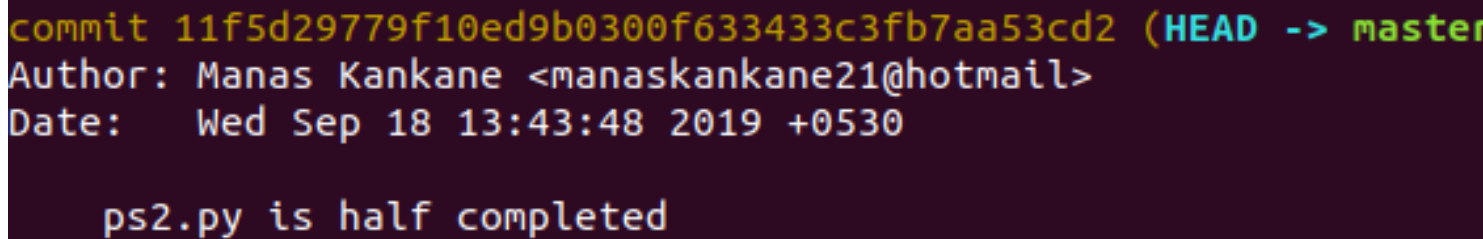
### 3 GIT Screenshots

A terminal screenshot showing a git commit command. The commit hash is a1c50da6df874a230fc4497d11df20b8699bba1b, and the message is 'ps1.py completed'. The author is Manas Kankane and the date is Wed Sep 18 10:29:39 2019 +0530. The HEAD points to the master branch.

```
commit a1c50da6df874a230fc4497d11df20b8699bba1b (HEAD -> master)
Author: Manas Kankane <manaskankane21@hotmail>
Date:   Wed Sep 18 10:29:39 2019 +0530

    ps1.py completed
```

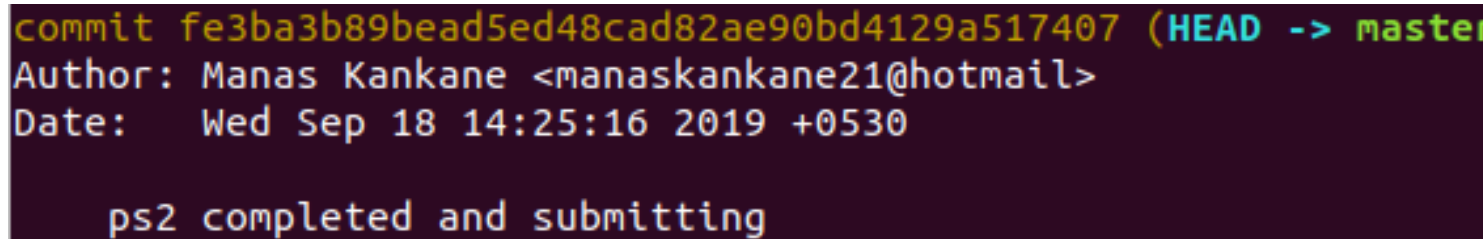
Figure 4: Screenshot of git1

A terminal screenshot showing a git commit command. The commit hash is 11f5d29779f10ed9b0300f633433c3fb7aa53cd2, and the message is 'ps2.py is half completed'. The author is Manas Kankane and the date is Wed Sep 18 13:43:48 2019 +0530. The HEAD points to the master branch.

```
commit 11f5d29779f10ed9b0300f633433c3fb7aa53cd2 (HEAD -> master)
Author: Manas Kankane <manaskankane21@hotmail>
Date:   Wed Sep 18 13:43:48 2019 +0530

    ps2.py is half completed
```

Figure 5: Screenshot of git2

A terminal screenshot showing a git commit command. The commit hash is fe3ba3b89bead5ed48cad82ae90bd4129a517407, and the message is 'ps2 completed and submitting'. The author is Manas Kankane and the date is Wed Sep 18 14:25:16 2019 +0530. The HEAD points to the master branch.

```
commit fe3ba3b89bead5ed48cad82ae90bd4129a517407 (HEAD -> master)
Author: Manas Kankane <manaskankane21@hotmail>
Date:   Wed Sep 18 14:25:16 2019 +0530

    ps2 completed and submitting
```

Figure 6: Screenshot of git3

## 4 Appendix

### 4.1 Appendix-ps1

```

1 a=input('Enter the parity bit data : ')
2 name=a.split()
3 b=0
4 c=0
5 x=[]
6 y=[]
7 for i in a:
8     if i=='1':
9         b=b+1
10    else:
11        c=0
12        if b%2==0:
13            for i in name:
14                name1=i + '1'
15            else:
16                for i in name:
17                    name1 = i + '0'
18            print('\nParity bit data is {}'.format(name1))
19            for i in name1:
20                x.append(i)
21            #print(x)
22            while c < len(a):
23                if x[c]=='0':
24                    if x[c+1]=='1':
25                        if x[c+2]=='0':
26                            x.insert(c+3,'1')
27                            y.append(c+3)
28                            c=c+1
29                            x[y[0]]='0'
30                            x[y[1]]='0'
31                            x[y[2]+1]='0'
32                            x.append('0101')
33                            #print(x)
34                            name2=''.join(x)
35                            print('\nTransmitting data : {}'.format(name2))

```

### 4.2 Appendix-ps2

```

1 print('_____')
2 print('Welcome to the Game')
3 print('_____')
4 list1=['0','0','0','0','0','0','0','0','0','0']
5 list2=[0,0,0,0,0,0,0,0,0,0]
6 a1=0
7 a2=0
8 a3=0
9 a4=0
10 a5=0
11 a6=0
12 a7=0
13 a8=0
14 a9=0
15 x=0
16 while 1:

```

```

17 print('\n1. Play Tic-Tac-Toe Game\n2. Exit')
18 a=int(input('\nEnter your choice : '))
19 if a==1:
20 b=str(input('\nEnter player 1 name : '))
21 c=str(input('\nEnter player 2 name : '))
22 d=str(input('\nWhat you want to become even player or odd player : '))
23 print('\nPlayer {} is odd player and Player {} is even player'.format(b,c))
24 print('\n-----Game Started-----\n')
25 string=''.join(list1)
26 print(string[0:3])
27 print(string[3:6])
28 print(string[6:9])
29 while x<=8:
30 pos1=input('Enter position player {} : '.format(b))
31 num1=input('Enter number player {} : '.format(b))
32 if num1 >='1' and num1 <='9':
33 if pos1=='1':
34 list1[0]=num1
35 list2[0]=int(num1)
36 string = ''.join(list1)
37 print(string[0:3])
38 print(string[3:6])
39 print(string[6:9])
40 a1=list2[0]
41 a2=list2[1]
42 a3 = list2[2]
43 a4 = list2[3]
44 a5 = list2[4]
45 a6 = list2[5]
46 a7 = list2[6]
47 a8 = list2[7]
48 a9 = list2[8]
49 if (a1+a2+a3)==15 or (a4+a5+a6)==15 or (a7+a8+a9)==15 or (a1+a4+a7)==15 or (a2+a5+
    a8)==15 or (a3+a6+a9)==15 or (a1+a5+a9)==15 or (a3+a5+a7)==15:
50 print('Player {} Wins'.format(b))
51 break
52 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
53 print('\n-----Game Draw-----\n')
54 print('\n-----Better Luck Next Time-----\n')
55 break
56 elif pos1=='2':
57 list1[1] = num1
58 list2[1] = int(num1)
59 string = ''.join(list1)
60 print(string[0:3])
61 print(string[3:6])
62 print(string[6:9])
63 a1 = list2[0]
64 a2 = list2[1]
65 a3 = list2[2]
66 a4 = list2[3]
67 a5 = list2[4]
68 a6 = list2[5]
69 a7 = list2[6]
70 a8 = list2[7]
71 a9 = list2[8]
72 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9

```

```

    ) == 15 or (a3 + a5 + a7) == 15:
73 print('Player {} Wins'.format(b))
74 break
75 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (
76 a1 + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (
77 a1 + a5 + a9) >= 15 and (a3 + a5 + a7) >= 15:
78 print('\n————Game Draw————\n')
79 print('\n————Better Luck Next Time————\n')
80 break
81 elif pos1=='3':
82 list1[2] = num1
83 list2[2] = int(num1)
84 string = ''.join(list1)
85 print(string[0:3])
86 print(string[3:6])
87 print(string[6:9])
88 a1 = list2[0]
89 a2 = list2[1]
90 a3 = list2[2]
91 a4 = list2[3]
92 a5 = list2[4]
93 a6 = list2[5]
94 a7 = list2[6]
95 a8 = list2[7]
96 a9 = list2[8]
97 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
    ) == 15 or (a3 + a5 + a7) == 15:
98 print('Player {} Wins'.format(b))
99 break
100 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (
101 a1 + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (
102 a1 + a5 + a9) >= 15 and (a3 + a5 + a7) >= 15:
103 print('\n————Game Draw————\n')
104 print('\n————Better Luck Next Time————\n')
105 break
106 elif pos1=='4':
107 list1[3] = num1
108 list2[3] = int(num1)
109 string = ''.join(list1)
110 print(string[0:3])
111 print(string[3:6])
112 print(string[6:9])
113 a1 = list2[0]
114 a2 = list2[1]
115 a3 = list2[2]
116 a4 = list2[3]
117 a5 = list2[4]
118 a6 = list2[5]
119 a7 = list2[6]
120 a8 = list2[7]
121 a9 = list2[8]
122 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
    ) == 15 or (a3 + a5 + a7) == 15:
123 print('Player {} Wins'.format(b))
124 break
125 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (
126 a1 + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (
127 a1 + a5 + a9) >= 15 and (a3 + a5 + a7) >= 15:

```

```

128 print('\n————Game Draw————\n')
129 print('\n————Better Luck Next Time————\n')
130 break
131 elif pos1=='5':
132     list1[4] = num1
133     list2[4] = int(num1)
134     string = ''.join(list1)
135     print(string[0:3])
136     print(string[3:6])
137     print(string[6:9])
138     a1 = list2[0]
139     a2 = list2[1]
140     a3 = list2[2]
141     a4 = list2[3]
142     a5 = list2[4]
143     a6 = list2[5]
144     a7 = list2[6]
145     a8 = list2[7]
146     a9 = list2[8]
147     if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
        a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
        ) == 15 or (a3 + a5 + a7) == 15:
148     print('Player {} Wins'.format(b))
149     break
150     if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (
151     a1 + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (
152     a1 + a5 + a9) >= 15 and (a3 + a5 + a7) >= 15:
153     print('\n————Game Draw————\n')
154     print('\n————Better Luck Next Time————\n')
155     break
156     elif pos1=='6':
157     list1[5] = num1
158     list2[5] = int(num1)
159     string = ''.join(list1)
160     print(string[0:3])
161     print(string[3:6])
162     print(string[6:9])
163     a1 = list2[0]
164     a2 = list2[1]
165     a3 = list2[2]
166     a4 = list2[3]
167     a5 = list2[4]
168     a6 = list2[5]
169     a7 = list2[6]
170     a8 = list2[7]
171     a9 = list2[8]
172     if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
        a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
        ) == 15 or (a3 + a5 + a7) == 15:
173     print('Player {} Wins'.format(b))
174     break
175     if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
        + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
        + a9) >= 15 and (a3 + a5 + a7) >= 15:
176     print('\n————Game Draw————\n')
177     print('\n————Better Luck Next Time————\n')
178     break
179     elif pos1=='7':
180     list1[6] = num1
181     list2[6] = int(num1)

```



```

182 string = ''.join(list1)
183 print(string[0:3])
184 print(string[3:6])
185 print(string[6:9])
186 a1 = list2[0]
187 a2 = list2[1]
188 a3 = list2[2]
189 a4 = list2[3]
190 a5 = list2[4]
191 a6 = list2[5]
192 a7 = list2[6]
193 a8 = list2[7]
194 a9 = list2[8]
195 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
    ) == 15 or (a3 + a5 + a7) == 15:
196 print('Player {} Wins'.format(b))
197 break
198 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
199 print('\n————Game Draw————\n')
200 print('\n————Better Luck Next Time————\n')
201 break
202 elif pos1=='8':
203 list1[7] = num1
204 list2[7] = int(num1)
205 string = ''.join(list1)
206 print(string[0:3])
207 print(string[3:6])
208 print(string[6:9])
209 a1 = list2[0]
210 a2 = list2[1]
211 a3 = list2[2]
212 a4 = list2[3]
213 a5 = list2[4]
214 a6 = list2[5]
215 a7 = list2[6]
216 a8 = list2[7]
217 a9 = list2[8]
218 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
    ) == 15 or (a3 + a5 + a7) == 15:
219 print('Player {} Wins'.format(b))
220 break
221 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
222 print('\n————Game Draw————\n')
223 print('\n————Better Luck Next Time————\n')
224 break
225 elif pos1=='9':
226 list1[8] = num1
227 list2[8] = int(num1)
228 string = ''.join(list1)
229 print(string[0:3])
230 print(string[3:6])
231 print(string[6:9])
232 a1 = list2[0]
233 a2 = list2[1]

```

```

234 a3 = list2[2]
235 a4 = list2[3]
236 a5 = list2[4]
237 a6 = list2[5]
238 a7 = list2[6]
239 a8 = list2[7]
240 a9 = list2[8]
241 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9
    ) == 15 or (a3 + a5 + a7) == 15:
242     print('Player {} Wins'.format(b))
243     break
244 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
245     print('\n-----Game Draw-----\n')
246     print('\n-----Better Luck Next Time-----\n')
247     break
248 else:
249     print('\n-----Game Error-----\n')
250     print('You entered wrong position')
251     print('\nEnter Again\n')
252     continue
253
254 pos2 = input('Enter position player {} : '.format(c))
255 num2 = input('Enter number player {} : '.format(c))
256 if num2 >= '1' and num2 <= '9':
257     if pos2 == '1':
258         list1[0] = num2
259         list2[0] = int(num2)
260         string = ''.join(list1)
261         print(string[0:3])
262         print(string[3:6])
263         print(string[6:9])
264         a1 = list2[0]
265         a2 = list2[1]
266         a3 = list2[2]
267         a4 = list2[3]
268         a5 = list2[4]
269         a6 = list2[5]
270         a7 = list2[6]
271         a8 = list2[7]
272         a9 = list2[8]
273         if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
            a4 + a7) == 15 or (
274         a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
            a7) == 15:
275             print('Player {} Wins'.format(c))
276             break
277         if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
            + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
            + a9) >= 15 and (a3 + a5 + a7) >= 15:
278             print('\n-----Game Draw-----\n')
279             print('\n-----Better Luck Next Time-----\n')
280             break
281         elif pos2 == '2':
282             list1[1] = num2
283             list2[1] = int(num2)
284             string = ''.join(list1)
285             print(string[0:3])

```

```

286 print(string[3:6])
287 print(string[6:9])
288 a1 = list2[0]
289 a2 = list2[1]
290 a3 = list2[2]
291 a4 = list2[3]
292 a5 = list2[4]
293 a6 = list2[5]
294 a7 = list2[6]
295 a8 = list2[7]
296 a9 = list2[8]
297 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
298 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
299 print('Player {} Wins'.format(c))
300 break
301 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
302 print('\n————Game Draw————\n')
303 print('\n————Better Luck Next Time————\n')
304 break
305 elif pos2 == '3':
306 list1[1] = num2
307 list2[1] = int(num2)
308 string = ''.join(list1)
309 print(string[0:3])
310 print(string[3:6])
311 print(string[6:9])
312 a1 = list2[0]
313 a2 = list2[1]
314 a3 = list2[2]
315 a4 = list2[3]
316 a5 = list2[4]
317 a6 = list2[5]
318 a7 = list2[6]
319 a8 = list2[7]
320 a9 = list2[8]
321 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
322 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
323 print('Player {} Wins'.format(c))
324 break
325 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
326 print('\n————Game Draw————\n')
327 print('\n————Better Luck Next Time————\n')
328 break
329 elif pos2 == '4':
330 list1[1] = num2
331 list2[1] = int(num2)
332 string = ''.join(list1)
333 print(string[0:3])
334 print(string[3:6])
335 print(string[6:9])
336 a1 = list2[0]
337 a2 = list2[1]

```

```

338 a3 = list2[2]
339 a4 = list2[3]
340 a5 = list2[4]
341 a6 = list2[5]
342 a7 = list2[6]
343 a8 = list2[7]
344 a9 = list2[8]
345 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
346 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
347 print('Player {} Wins'.format(c))
348 break
349 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
350 print('\n-----Game Draw-----\n')
351 print('\n-----Better Luck Next Time-----\n')
352 break
353 elif pos2 == '5':
354 list1[1] = num2
355 list2[1] = int(num2)
356 string = ''.join(list1)
357 print(string[0:3])
358 print(string[3:6])
359 print(string[6:9])
360 a1 = list2[0]
361 a2 = list2[1]
362 a3 = list2[2]
363 a4 = list2[3]
364 a5 = list2[4]
365 a6 = list2[5]
366 a7 = list2[6]
367 a8 = list2[7]
368 a9 = list2[8]
369 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
370 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
371 print('Player {} Wins'.format(c))
372 break
373 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
374 print('\n-----Game Draw-----\n')
375 print('\n-----Better Luck Next Time-----\n')
376 break
377 elif pos2 == '6':
378 list1[1] = num2
379 list2[1] = int(num2)
380 string = ''.join(list1)
381 print(string[0:3])
382 print(string[3:6])
383 print(string[6:9])
384 a1 = list2[0]
385 a2 = list2[1]
386 a3 = list2[2]
387 a4 = list2[3]
388 a5 = list2[4]
389 a6 = list2[5]

```

```

390 a7 = list2[6]
391 a8 = list2[7]
392 a9 = list2[8]
393 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
394 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
395 print('Player {} Wins'.format(c))
396 break
397 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
398 print('\n————Game Draw————\n')
399 print('\n————Better Luck Next Time————\n')
400 break
401 elif pos2 == '7':
402 list1[1] = num2
403 list2[1] = int(num2)
404 string = ''.join(list1)
405 print(string[0:3])
406 print(string[3:6])
407 print(string[6:9])
408 a1 = list2[0]
409 a2 = list2[1]
410 a3 = list2[2]
411 a4 = list2[3]
412 a5 = list2[4]
413 a6 = list2[5]
414 a7 = list2[6]
415 a8 = list2[7]
416 a9 = list2[8]
417 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
418 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
419 print('Player {} Wins'.format(c))
420 break
421 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
422 print('\n————Game Draw————\n')
423 print('\n————Better Luck Next Time————\n')
424 break
425 elif pos2 == '8':
426 list1[1] = num2
427 list2[1] = int(num2)
428 string = ''.join(list1)
429 print(string[0:3])
430 print(string[3:6])
431 print(string[6:9])
432 a1 = list2[0]
433 a2 = list2[1]
434 a3 = list2[2]
435 a4 = list2[3]
436 a5 = list2[4]
437 a6 = list2[5]
438 a7 = list2[6]
439 a8 = list2[7]
440 a9 = list2[8]
441 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +

```

```

    a4 + a7) == 15 or (
442 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
443 print('Player {} Wins'.format(c))
444 break
445 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
446 print('\n-----Game Draw-----\n')
447 print('\n-----Better Luck Next Time-----\n')
448 break
449 elif pos2 == '9':
450 list1[1] = num2
451 list2[1] = int(num2)
452 string = ''.join(list1)
453 print(string[0:3])
454 print(string[3:6])
455 print(string[6:9])
456 a1 = list2[0]
457 a2 = list2[1]
458 a3 = list2[2]
459 a4 = list2[3]
460 a5 = list2[4]
461 a6 = list2[5]
462 a7 = list2[6]
463 a8 = list2[7]
464 a9 = list2[8]
465 if (a1 + a2 + a3) == 15 or (a4 + a5 + a6) == 15 or (a7 + a8 + a9) == 15 or (a1 +
    a4 + a7) == 15 or (
466 a2 + a5 + a8) == 15 or (a3 + a6 + a9) == 15 or (a1 + a5 + a9) == 15 or (a3 + a5 +
    a7) == 15:
467 print('Player {} Wins'.format(c))
468 break
469 if (a1 + a2 + a3) >= 15 and (a4 + a5 + a6) >= 15 and (a7 + a8 + a9) >= 15 and (a1
    + a4 + a7) >= 15 and (a2 + a5 + a8) >= 15 and (a3 + a6 + a9) >= 15 and (a1 + a5
    + a9) >= 15 and (a3 + a5 + a7) >= 15:
470 print('\n-----Game Draw-----\n')
471 print('\n-----Better Luck Next Time-----\n')
472 break
473 else:
474 print('\n-----Game Error-----\n')
475 print('You entered wrong position')
476 print('\nEnter Again\n')
477 continue
478
479 x=x+1
480 else:
481 print('_____')
482 print('Thank You for Playing Game')
483 print('_____')
484 exit()

```

## References

- [1] Basic Syntex. *tutorials point*. [https://www.tutorialspoint.com/python/python\\_command\\_line\\_arguments.htm](https://www.tutorialspoint.com/python/python_command_line_arguments.htm).
- [2] For seeing functions. *W3school*. [https://www.w3schools.com/python/python\\_ref\\_functions.asp](https://www.w3schools.com/python/python_ref_functions.asp).
- [3] Inserting into a list. *GeeksforGeeks*. <https://www.geeksforgeeks.org/python-list-insert/>.
- [4] Tic-Tak-Toe Game. *GeeksforGeeks*. <https://www.geeksforgeeks.org/python-implementation-automatic-tic-tac-toe-game-using-random-number/>.