

Index

Srno	Date	Practical	Sign
1	8-11-23	Practical - 1	{ JKT }
2	29-11-23	Practical - 2	{ JKT }
3	15-12-23	Practical - 3	{ JKT }
4	23-12-23	Practical - 4	{ JKT }
5	4-1-24	Practical - 5	{ JKT }
6	13-1-24	Practical - 6	{ JKT }
7	4-15-1-24	Practical - 7	{ JKT }
8	29-01-24	Practical - 8	{ JKT }
9	12-02-24	Practical - 9	

Complete
12/02/24



CLASS : _____

DATE : _____

Practical No. : 1

Topic : _____

write a program for tokenisation of the
given input

- ▶ pip install nltk
- ▶ import nltk
- ▶ nltk.download('all')
- import nltk.data
- ▶ from nltk.tokenize import sent_tokenize
text = "Hello everyone everyone welcome to sies"
sentence = sent_tokenize(text)
print(sentence)

O/p ['Hello', 'everyone', 'everyone', 'welcome', 'to', 'sies']

- ▶ from nltk.tokenize import word_tokenize
text = "Hello everyone welcome to sies"
sentence = word_tokenize(text)
print(sentence)

O/p ['Hello', 'everyone', 'welcome', 'to', 'sies']

• `tokenizer = nltk.data.load('tokenizers/punkt
english.pickle')`
`tokenizer.tokenize(text)`

Output: ['Hello everyone welcome to sies']

• `spanish_tokenizer = nltk.data.load('tokenizers/
punkt/spanish.pickle')`
`text = 'holo amigo. Estoy bien'`
`tokenizer.tokenize(text)`

Output: ['holo amigo.', 'Estoy bien']

~~Amigo~~
~~Spanish~~
comes



SIES (Nerul) College of Arts, Science and Commerce

UGC Recognized under 2(F) and 12(B) UGC ACT 1956.
NAAC Reaccredited - "A" Grade. Affiliated to University of Mumbai
PLOT NO. 1-C, SECTOR-V, NERUL, NAVI MUMBAI - 400 706.



(Nerul) College of Arts,
Science & Commerce
(Autonomous)

CLASS : _____

DATE : 1 / 1 / 24

Practical No.: TOC - 2 Topic : _____

import re

line = "horses are taller than dogs";

searchObj = re.search(r'(.*) are (.*)', line, re.M | re.I)

if searchObj :

print("searchObj.group():", searchObj.group())

print("searchObj.group(1):", searchObj.group(1))

print("searchObj.group(2):", searchObj.group(2))

else:

print("Nothing found!!")

O/P

searchObj.group(): horses are taller than dogs

searchObj.group(1) : horses

searchObj.group(2) : taller

X
Complete



CLASS : _____

DATE : _____

Practical No.: 3 Topic : _____

Q. Write a program for generating derivation sequence / language for the given sequence of production

```
def printArray(arr, size):  
    for i in range(size):  
        print(arr[i], end=" ")  
    print()  
    return
```

```
def getSuccessor(arr, K, n):  
    p = K - 1  
    while arr[p] == n and 0 <= p < K:  
        p -= 1  
    if (p < 0):  
        return 0  
    arr[p] = arr[p] + 1  
    i = p + 1  
    while (i < K):  
        arr[i] = 1  
        i += 1  
    return 1
```

```
def printSequence(n, k):  
    arr = [0]*k  
    for i in range(k):  
        arr[i] = 1  
    while True:
```

Page No. _____

point Array($a[0:n], k$)
 if getSuccessor($a[0:n], k, n$) = = 6,
 break
 return

 $n = 4$ $k = 2$ point Sequence(n, k) $n = 2$ $k = 2$ point Sequence(n, k)

O/P

1 1

1 2

1 3

1 4

2 1

2 2

2 3

2 4

3 1

3 2

3 3

3 4

4 1

4 2

4 3

4 4

O/P

1 1

1 2

2 1

2 2

 $n = 2$ $k = 3$

O/P

1 1 1

1 1 2

1 2 1

1 2 2

2 1 1

2 1 2

2 2 1

2 2 2

~~Conjugated~~
~~an array~~
~~of size~~
~~(n-k)~~



CLASS : _____

DATE : 23-12-23

Practical No.: 4 Topic : ~~Design a Program for creating machine that accepts three consecutive one~~

#State A

```
def stateA(n):
    if (n[0] == 'a'):
        stateB(n[1:])
    elif (n[0] == 'b'):
        stateH(n[1:])
```

State B

```
def stateB(n):
    if (len(n) == 0):
        print("string Not Accepted")
    else:
        if (n[0] == 'a'):
            stateC(n[1:])
        elif (n[0] == 'b'):
            stateI(n[1:]))
```

def stateC(n):

```
if (len(n) == 0):
    print("string Not Accepted")
```

else :

```
if (n[0] == 'a'):
```

```
stateD(n[1:]))
```

```
elif (n[0] == 'b'):
```

```
stateJ(n[1:]))
```

Page No. _____

fig - 1

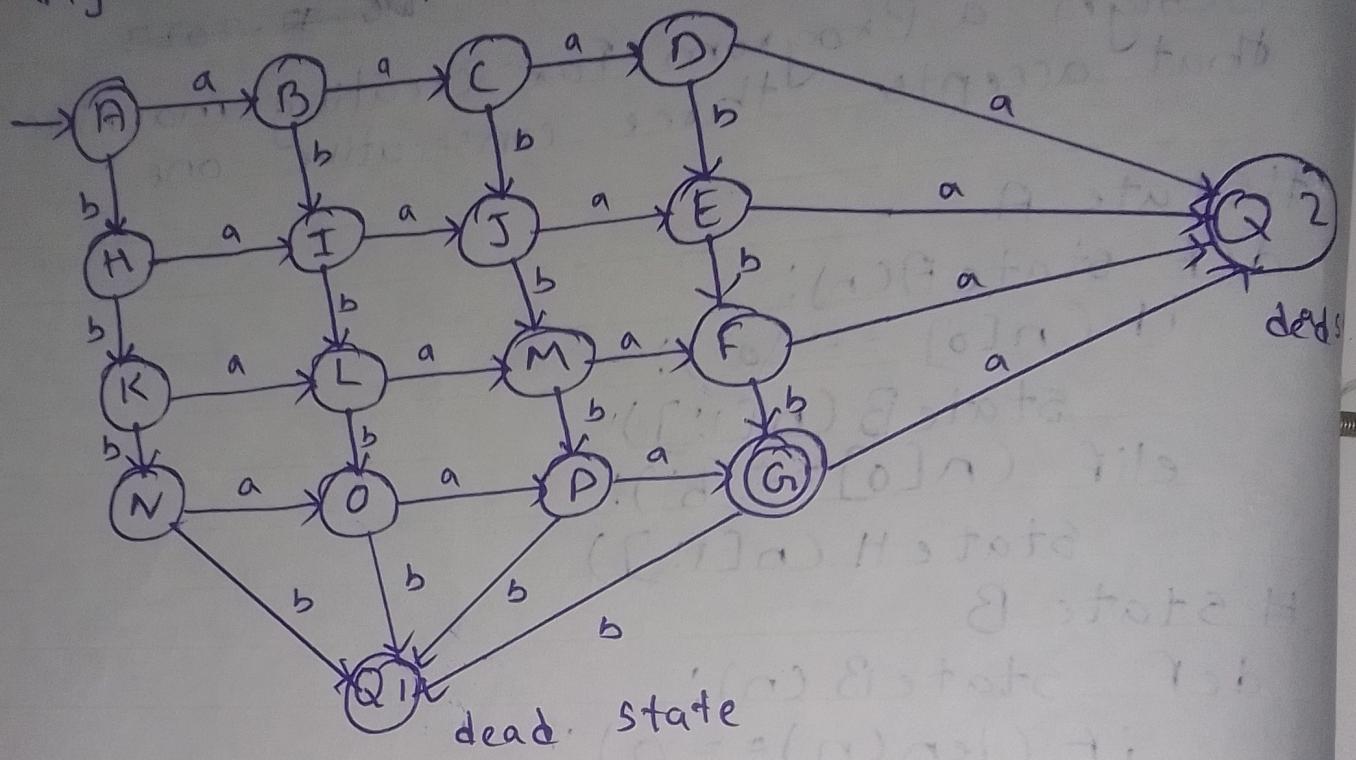
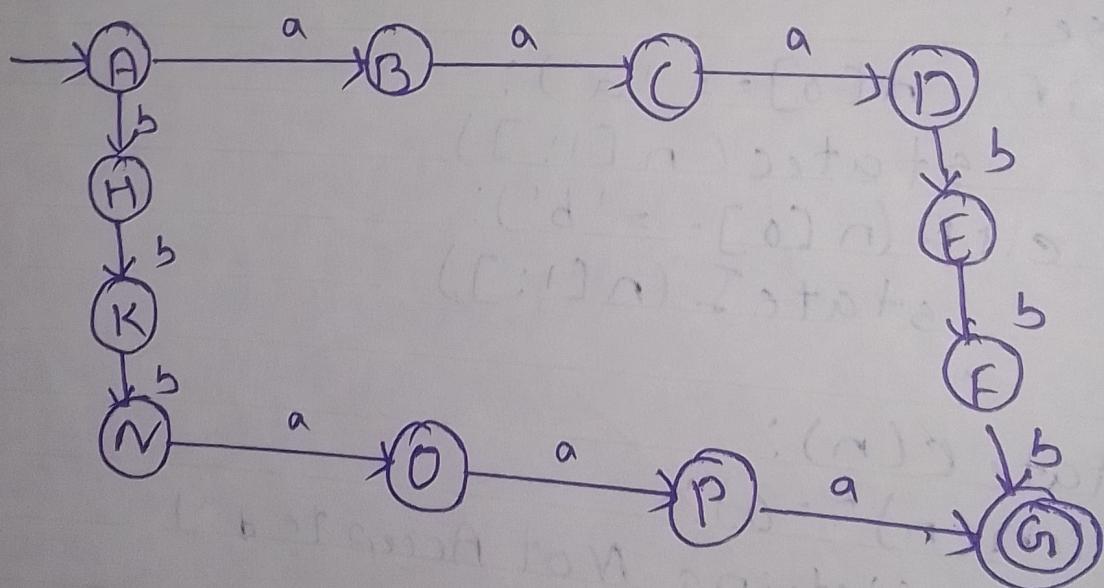


fig 2:



```

def state D(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 0):
            print ("String Not Accepted")
        else:
            if (n[0] == 'a'):
                state Q7(n)
            elif (n[0] == 'b'):
                state E(n[1:])

```

```

def state E(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state Q2(n)
        elif (n[0] == 'b'):
            state F(n[1:])

```

```

def state F(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state Q2 (n[1:])
        elif (n[0] == 'b'):
            state G(n[1:])

```

```

def state G(n):
    if (len(n) == 0):
        print("String Accepted")
    else:
        if (n[0] == 'a'):
            state Q2(n)
        elif (n[0] == 'b'):
            state Q2(n)

```

```

def state H(n):
    if (len(n) == 0):
        print("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state J(n[1:])
        elif (n[0] == 'b'):
            state K(n[1:])

```

```

def state I(n):
    if (len(n) == 0):
        print("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state J(n[1:])
        elif (n[0] == 'b'):
            state L(n[1:])

```

```

def stateS(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state E(n[1:])
        elif (n[0] == 'b'):
            state M(n[1:])

```

```

def stateK(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state L(n[1:])
        elif (n[0] == 'b'):
            state N(n[1:])

```

```

def stateL(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state M(n[1:])
        elif (n[0] == 'b'):
            state G(n[1:])

```

```

def state M(n):
    if (len(n) == 0):
        print("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state F(n[1:])
        elif (n[0] == 'b'):
            state P(n[1:])

```

```

def State N(n):
    if (len(n) == 0):
        print ("String Not Accepted")
    else:
        if (n[0] == 0):
            state O(n[1:])
        elif (n[0] == 'b'):
            state Q1(n)

```

```

def state O(n):
    if (len(n) == 0):
        print("String Not Accepted")
    else:
        if (n[0] == 'a'):
            state P(n[1:])
        elif (n[0] == 'b'):
            state Q1(n)

```

```

def stateP(n):
    if (len(n) == 0):
        print("String Not Accepted")
    else:
        if (n[0] == 'a'):
            stateG(n[1:])
        elif (n[0] == 'b'):
            stateQ1(n[1:])

```

```

def stateQ1(n):
    print("String Not Accepted")

```

```

def stateQ2(n):
    print("String not Accepted")

```

~~n = "aaabbb"~~

~~stateA(n)~~

^{2S1/2(B)}

~~Accepted~~

~~Completed~~



CLASS : _____

DATE : 1/1/29

Practical No.: TOC - P5 Topic : _____

Design a Program to creating machine that accepts the string always ends with 101

```
def q1(s, i):
    print("q1->", end=" ")
    if (i == len(s)):
        print("No")
        return
    if (s[i] == '0'):
        q1(s, i+1)
    else:
        q3(s, i+1)
```

```
def q2(s, i):
    print("q2->", end=" ")
    if (i == len(s)):
        print("No")
        return
    if (s[i] == '0'):
        q4(s, i+1)
    else:
        q2(s, i+1)
```

```
def q3(s, i):
    print("q3->", end=" ")
    if (i == len(s)):
```

Page No. _____

```
print("YES")
return
if s[i] == '0':
    q4(s, i+1)
else:
```

```
    q2(s, i+1)
    print("NO")  
    return
def q4(s, i):
    print("q4->", end=" ")
    if i == len(s):
        print("end")
        print("YES")
        return
    if s[i] == '0':
        q1(s, i+1)
    else:
        q3(s, i+1)
```

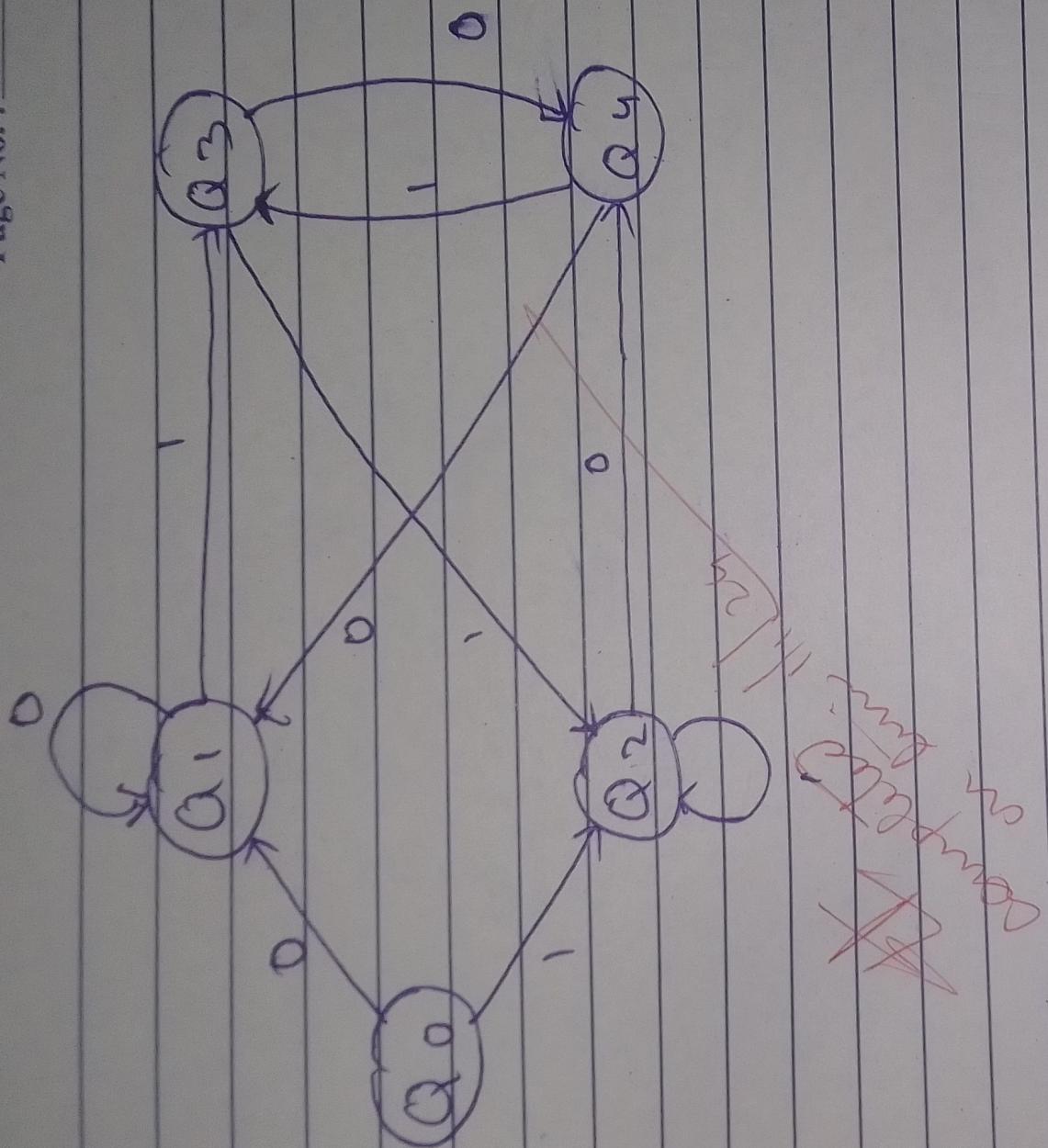
```
def q0(s, i):
    print("q0->", end=" ")
    if i == len(s):
        print("end")
        print("NO")
        return
    if s[i] == '0':
        q1(s, i+1)
    else:
        q2(s, i+1)
```

s = "1010"

print("State transition are", end=" ")
q0(s, 0)

O/p

State transitions are q0 -> q2 -> q4 -> q3 -> q4 -> YES





SIES (Nerul) College of Arts, Science and Commerce

UGC Recognized under 2(f) and 12(B) UGC ACT 1956.

NAAC Reaccredited - A Grade Affiliated to University of Mumbai

PLOT NO. 1-C, SECTOR V, NERUL, NAVI MUMBAI - 400 706,

CLASS :

DATE : 13/01/2023

Practical No.: 6

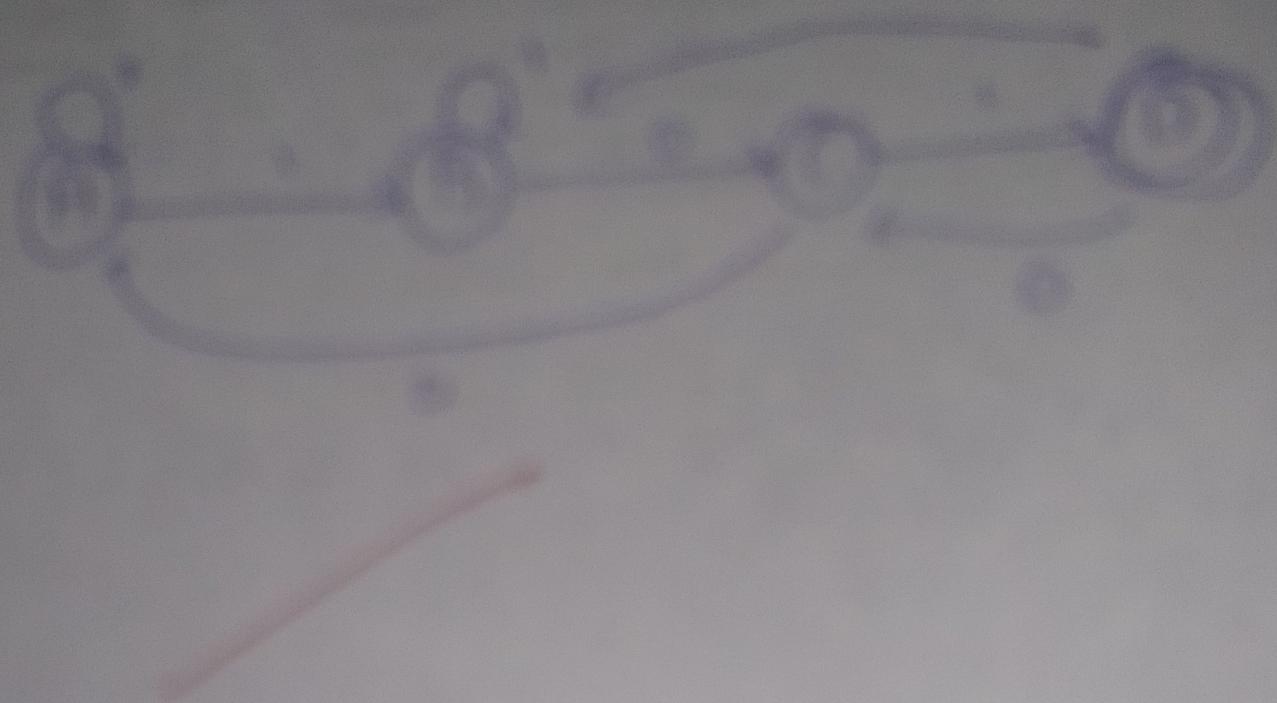
Topic :

Write a program to develop a machine
that ends with 101

```
def q1(s,i):
    print("q1->" , end = " ")
    if i == len(s):
        print("No")
        return
    if s[i] == '0':
        q1(s,i+1)
    else:
        q2(s,i+1)
```

```
def q2(s,i):
    print("q2->" , end = " ")
    if i == len(s):
        print("No")
        return
    if s[i] == '0':
        q3(s,i+1)
    else:
        q2(s,i+1)
```

```
def q3(s,i):
    print("q3->" , end = " ")
    if i == len(s):
        print("No")
```



```

print("YES")
return
if s[i] == '0':
    q1(s, i+1)
else:
    q4(s, i+1)

```

```

def q4(s, i):
    print("q4->, end=' ")
    if i == len(s):
        print("YES")
    return
    if s[i] == '0':
        q3(s, i+1)
    else:
        q2(s, i+1)

```

$s = "1110"$

print("State transitions are", end=')

q1(s, 0)

O/P

$q1 \rightarrow q2 \rightarrow q2 \rightarrow q3 \rightarrow q3 \rightarrow YES$

$s = "101"$

O/P:

$q1 \rightarrow q2 \rightarrow q3 \rightarrow q4$

$s = "010"$

O/P $q1 \rightarrow q1 \rightarrow q2 \rightarrow q3 \rightarrow YES$

$s = "$

~~Can't find any~~

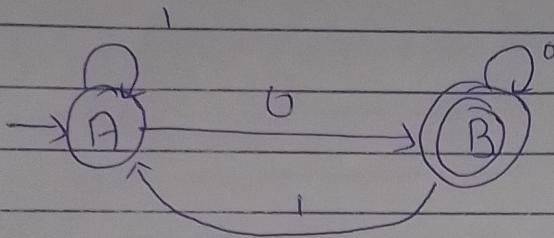


CLASS : _____

DATE : 15-01-24

Practical No.: 87 Topic : _____

Design a program for accepting decimal number divisible by 2



Program

```
def stateA(n):
    if (len(n) == 0):
        print ("string accepted => YES")
    else:
        if (n[0] == '0'):
            stateA(n[1:])
        elif (n[0] == '1'):
            stateB(n[1:])

def stateB(n):
    if (len(n) == 0):
        print ("string not accepted => NO")
    else:
        if (n[0] == '0'):
            stateA(n[1:])
        elif (n[0] == '1'):
            stateB(n[1:])
```

$n = \text{int}(\text{input}())$

$n = \text{bin}(n).replace('0b', '')$

$\text{stateBN}(n)$

O/p

$n = 15$

string not accepted \Rightarrow NO

O/p

$n = 10$

string accepted \Rightarrow YES

O/p

$n = 100$

string accepted \Rightarrow YES

X completed 18/12/24
Complexity



SIES (Nerul) College of Arts, Science and Commerce

UGC Recognized under 2(F) and 12(B) UGC ACT 1956.
NAAC Accredited "A" Grade, Affiliated to University of Mumbai
PLOT NO. 1-C, SECTOR-V, NERUL, NAVI MUMBAI - 400 706.

(Nerul) College of Arts,
Science & Commerce
SIES
(Autonomous)
RISE WITH EDUCATION

CLASS : _____

DATE : _____

Practical No.: _____

Topic : _____

Design a program for creating a machine which
accepts string having equal no. of 1's, 2's & 0's

def getSubStringWithEqual012(s):

arr = [];

n = len(s);

for i in range(n):

 for j in range(i, n):

 s1 = s[i:j];

 arr.append(s1);

 count = 0;

 for k in range(len(arr)):

 if arr[k] == s1:

 count += 1;

 if count == 0:

 count += 1;

 if count == 1:

 count += 1;

 if count == 2:

 count += 1;

 if count == 3:

 count += 1;

 if count == 4:

 count += 1;

 if count == 5:

 count += 1;

 if count == 6:

 count += 1;

```
if (countZero == countOne & countOnes == countTwo)  
    count += 1;  
return count;
```

```
str = "0102010";  
print (getSubStringWithEqual1012(str));
```

O/p 2

~~XX~~



SIES (Nerul) College of Arts, Science and Commerce

UGC Recognized under 2(F) and 12(B) UGC ACT 1956,
NAAC Reaccredited - "A" Grade, Affiliated to University of Mumbai
PLOT NO. 1-C, SECTOR-V, NERUL, NAVI MUMBAI - 400 706.



(Nerul) College of Arts,
Sciences & Commerce
(Autonomous)

SIES
RISE WITH EDUCATION

CLASS : _____

DATE : _____

Practical No.: 9 Topic : _____

Design a program for a creating a machine which
count number of 1's and 0's in a given string

```
def countsubstring(s, n):
    ans = 0;
    i = 0;
    while (i < n):
        cnt0 = 0; cnt1 = 0;
        if (s[i] == '0'):
            while (i < n and s[i] == '0'):
                cnt0 += 1;
                i += 1;
                i += 1;
                j = i;
            while (j < n and s[j] == ','):
                cnt1 += 1;
                j += 1;
        else:
            while (i < n and s[i] == '1'):
                cnt1 += 1;
                i += 1;
                i += 1;
                j = i;
            while (j < n and s[j] == ','):
                cnt0 += 1;
                j += 1;
    ans += min(cnt0, cnt1)
```

ans += min(cnt0, cnt1) Page No. _____

```

return ans;
if name == "main":
    s = "0001110010";
n = len(s);
print(countSubString(s, n));

```

~~Ques 2020~~

OR

```

cnt = 0
def count_zero(s: str):
    global cnt
    for i in range(len(s)):
        div = int(s[i])
        if div == 0:
            cnt += 1

```

~~print(
print(cnt))~~

```

s = "101101"
count_zero(s)

```

O/p

2

~~XX after 12th M
Campus visit~~