NAME: PATEL JAIMINI SHAILESHKUMAR

ROLLNO: 37

SYMCA-SEM-IV

SUBJECT: IMAGE PROCESSING AND COMPUTER VISION

Roll no: 37
Name: jaimini patel

Ex no: 1

Date: 28-11-2019

Problem Statement:

Write a Python graphics program to print your fullname in English using basic shapes.

Program:

```
from graphics import *
w=GraphWin("alpha",1000,700)
l=Line(Point(0,10),Point(1000,10))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(0,650),Point(1000,650))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#J
l=Line(Point(10,100),Point(200,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(100,200),Point(100,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Arc(Point(100,250),Point(30,150),0,-180,"arc")
l.draw(w)
l.setOutline("red")
```

```
l.setWidth(3)
#A
l=Line(Point(240,100),Point(200,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(240,100),Point(300,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(215,200),Point(280,200))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#I
l=Line(Point(320,100),Point(450,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(380,250),Point(380,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(320,250),Point(450,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#M
l=Line(Point(470,100),Point(470,255))
l.draw(w)
```

```
l.setWidth(3)
l.setOutline("red")
l=Line(Point(470,100),Point(520,170))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(570,100),Point(520,170))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(570,100),Point(570,255))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#I
l=Line(Point(590,100),Point(740,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(665,250),Point(665,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(590,250),Point(740,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#N
l=Line(Point(770,100),Point(770,255))
l.draw(w)
```

```
l.setWidth(3)
l.setOutline("red")
l=Line(Point(770,100),Point(850,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(850,100),Point(850,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#I
l=Line(Point(860,100),Point(1000,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(930,250),Point(930,100))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(860,250),Point(1000,250))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#P
l=Line(Point(70,550),Point(70,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Arc(Point(0,450),Point(150,350),90,-180,"arc")
l.draw(w)
```

```
l.setOutline("red")
l.setWidth(3)
#A
l=Line(Point(170,550),Point(220,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(280,550),Point(220,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(188,470),Point(255,470))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#T
l=Line(Point(480,350),Point(280,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(380,550),Point(380,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#E
l=Line(Point(520,550),Point(520,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(650,350),Point(520,350))
```

```
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(650,450),Point(520,450))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(650,550),Point(520,550))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
#L
l=Line(Point(690,550),Point(690,350))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(860,550),Point(690,550))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
```

Roll no: 37
Name: jaimini patel

Output:

Roll no: 37
Name: jaimini patel

Ex no: 2

Date: 28-11-2019

Problem Statement:

Write a Python graphics program to print your fullname in Hindi using basic shapes.

Program:

```
from graphics import *
w=GraphWin("Hindi Name",700,700)
l=Line(Point(0,10),Point(700,10))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(0,550),Point(700,550))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
w.setBackground('black')
#Jai
l=Line(Point(260,100),Point(550,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(280,145),Point(340,145))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(250,200),Point(290,110),10,-200,"arc")
l.draw(w)
```

```
l.setOutline("blue")
l.setWidth(3)
l=Line(Point(340,200),Point(340,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(250,45),Point(350,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(220,50),Point(320,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
#mi
l=Line(Point(430,200),Point(430,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(400,145),Point(430,145))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(400,170),Point(400,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(400,180),Point(370,140),0,-340,"arc")
l.draw(w)
l.setOutline("blue")
```

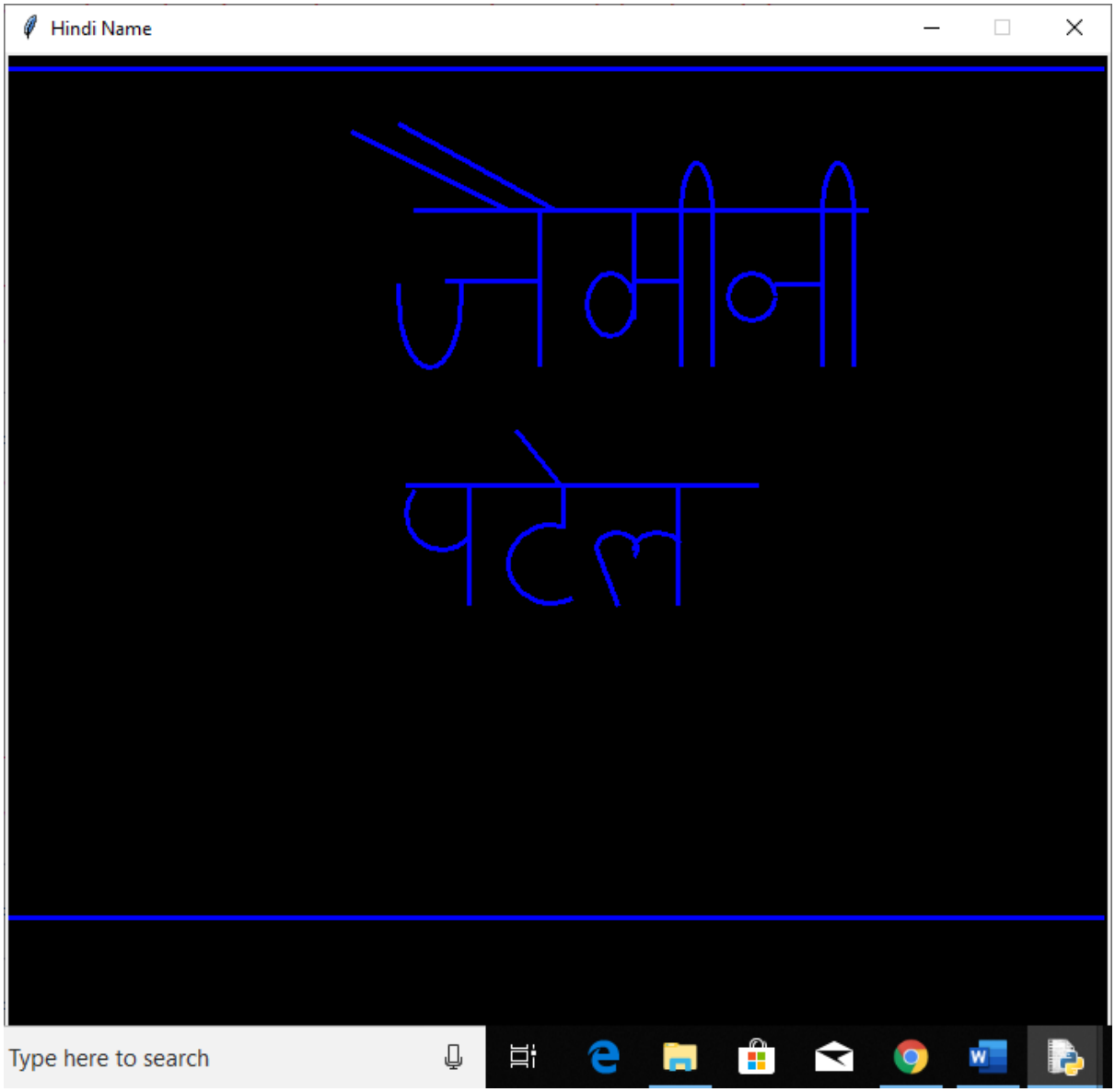
```
l.setWidth(3)
l=Line(Point(450,200),Point(450,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(450,130),Point(430,70),0,180,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
#ni
l=Line(Point(520,200),Point(520,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(490,147),Point(520,147))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(490,170),Point(460,140),0,355,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Line(Point(540,200),Point(540,100))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(540,130),Point(520,70),0,180,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
```

```
#p
l=Line(Point(255,275),Point(480,275))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Arc(Point(255,270),Point(300,316),140,182,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Line(Point(295,277),Point(295,352))
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
#t
l=Line(Point(355,277),Point(355,302))
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Arc(Point(320,300),Point(375,350),70,230,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Line(Point(355,277),Point(325,240))
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
#l
l=Line(Point(390,352),Point(376,315))
l.draw(w)
l.setOutline("blue")
```

```
l.setWidth(3)
l=Arc(Point(376,305),Point(402,325),-35,220,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Arc(Point(400,305),Point(430,325),20,170,"arc")
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
l=Line(Point(428,277),Point(428,352))
l.draw(w)
l.setOutline("blue")
l.setWidth(3)
```

Roll no: 37
Name: jaimini patel

Output:



Ex no: 3

Date: 28-11-2019

Problem Statement:

Write a Python graphics program to print alphabets in both upper case and lower case letters using basic shapes.

Capital letter

Program:

```
from graphics import *
w=GraphWin("Alphabets(A-Z)",600,500)
l=Line(Point(0,10),Point(600,10))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
l=Line(Point(0,380),Point(650,380))
l.draw(w)
l.setWidth(3)
l.setOutline("blue")
w.setBackground("black")
a1=Line(Point(22,120),Point(40,20))
a2=Line(Point(60,120),Point(40,20))
a3=Line(Point(30,80),Point(50,80))
a1.setWidth(3)
a2.setWidth(3)
a3.setWidth(3)
a1.draw(w)
a2.draw(w)
a3.draw(w)
a1.setOutline("red")
```

```
a2.setOutline("red")
a3.setOutline("red")
b1=Line(Point(80,20),Point(80,120))
b2=Arc(Point(45,20),Point(120,70),270,180,"arc")
b3=Arc(Point(45,70),Point(120,120),270,180,"arc")
b1.setWidth(3)
b1.draw(w)
b2.draw(w)
b3.draw(w)
b2.setWidth(4)
b3.setWidth(4)
b1.setOutline("red")
b2.setOutline("red")
b3.setOutline("red")
c1=Arc(Point(140,20),Point(190,120),45,270,"arc")
c1.draw(w)
c1.setWidth(3)
c1.setOutline("red")
d1=Line(Point(200,20),Point(200,120))
d2=Arc(Point(160,20),Point(240,120),270,180,"arc")
d2.draw(w)
d2.setWidth(3)
d1.setWidth(3)
d1.draw(w)
d1.setOutline("red")
d2.setOutline("red")
e1=Line(Point(260,20),Point(260,120))
e2=Line(Point(260,20),Point(300,20))
e3=Line(Point(260,70),Point(290,70))
e4=Line(Point(260,120),Point(300,120))
```

```
e1.setWidth(3)
e2.setWidth(3)
e3.setWidth(3)
e4.setWidth(3)
e1.draw(w)
e2.draw(w)
e3.draw(w)
e4.draw(w)
e1.setOutline("red")
e2.setOutline("red")
e3.setOutline("red")
e4.setOutline("red")
f1=Line(Point(320,20),Point(320,120))
f2=Line(Point(320,20),Point(360,20))
f3=Line(Point(320,70),Point(350,70))
f1.setWidth(3)
f2.setWidth(3)
f3.setWidth(3)
f1.draw(w)
f2.draw(w)
f3.draw(w)
f1.setOutline("red")
f2.setOutline("red")
f3.setOutline("red")
g1=Arc(Point(370,20),Point(420,120),45,300,"arc")
g2=Line(Point(400,80),Point(430,80))
g3=Line(Point(420,80),Point(420,120))
g1.draw(w)
g1.setWidth(3)
g2.setWidth(3)
```

```
g3.setWidth(3)
g2.draw(w)
g3.draw(w)
g1.setOutline("red")
g2.setOutline("red")
g3.setOutline("red")
h1=Line(Point(440,20),Point(440,120))
h2=Line(Point(440,70),Point(480,70))
h3=Line(Point(480,20),Point(480,120))
h1.setWidth(3)
h2.setWidth(3)
h3.setWidth(3)
h1.draw(w)
h2.draw(w)
h3.draw(w)
h1.setOutline("red")
h2.setOutline("red")
h3.setOutline("red")
i1=Line(Point(520,20),Point(520,120))
i2=Line(Point(510,20),Point(530,20))
i3=Line(Point(510,120),Point(530,120))
i1.setWidth(3)
i2.setWidth(3)
i3.setWidth(3)
i1.draw(w)
i2.draw(w)
i3.draw(w)
i1.setOutline("red")
i2.setOutline("red")
i3.setOutline("red")
```

```
j1=Line(Point(580,20),Point(580,115))
j2=Line(Point(560,20),Point(600,20))
j3=Arc(Point(555,100),Point(580,120),160,180,"arc")
j3.draw(w)
j3.setWidth(3)
j1.setWidth(3)
j2.setWidth(3)
j1.draw(w)
j2.draw(w)
j1.setOutline("red")
j2.setOutline("red")
j3.setOutline("red")
k1=Line(Point(20,140),Point(20,240))
k2=Line(Point(60,140),Point(20,190))
k3=Line(Point(20,190),Point(60,240))
k1.setWidth(3)
k2.setWidth(3)
k3.setWidth(3)
k1.draw(w)
k2.draw(w)
k3.draw(w)
k1.setOutline("red")
k2.setOutline("red")
k3.setOutline("red")
l=Line(Point(0,130),Point(650,130))
l.draw(w)
l.setWidth(3)
l.setOutline("pink")
```

```
l1=Line(Point(80,140),Point(80,240))
l2=Line(Point(80,240),Point(120,240))
l1.setWidth(3)
l2.setWidth(3)
l1.draw(w)
l2.draw(w)
l1.setOutline("red")
l2.setOutline("red")
m1=Line(Point(140,140),Point(140,240))
m2=Line(Point(140,140),Point(160,190))
m3=Line(Point(160,190),Point(180,140))
m4=Line(Point(180,140),Point(180,240))
m1.setWidth(3)
m2.setWidth(3)
m3.setWidth(3)
m4.setWidth(3)
m1.draw(w)
m2.draw(w)
m3.draw(w)
m4.draw(w)
m1.setOutline("red")
m2.setOutline("red")
m3.setOutline("red")
m4.setOutline("red")
n1=Line(Point(200,140),Point(200,240))
n2=Line(Point(200,140),Point(240,240))
n3=Line(Point(240,140),Point(240,240))
n1.setWidth(3)
n2.setWidth(3)
n3.setWidth(3)
```

```
n1.draw(w)
n2.draw(w)
n3.draw(w)
n1.setOutline("red")
n2.setOutline("red")
n3.setOutline("red")
o=Oval(Point(260,140),Point(300,240))
o.setWidth(3)
o.draw(w)
o.setOutline("red")
p1=Line(Point(320,140),Point(320,240))
p2=Arc(Point(285,140),Point(360,190),270,180,"arc")
p2.draw(w)
p2.setWidth(3)
p1.setWidth(3)
p1.draw(w)
p1.setOutline("red")
p2.setOutline("red")
q1=Oval(Point(370,140),Point(420,240))
q2=Line(Point(400,220),Point(420,250))
q1.setWidth(3)
q1.draw(w)
q2.setWidth(3)
q2.draw(w)
q1.setOutline("red")
q2.setOutline("red")
r1=Line(Point(440,140),Point(440,240))
r2=Arc(Point(405,140),Point(480,190),270,180,"arc")
r3=Line(Point(440,190),Point(480,240))
r2.draw(w)
```

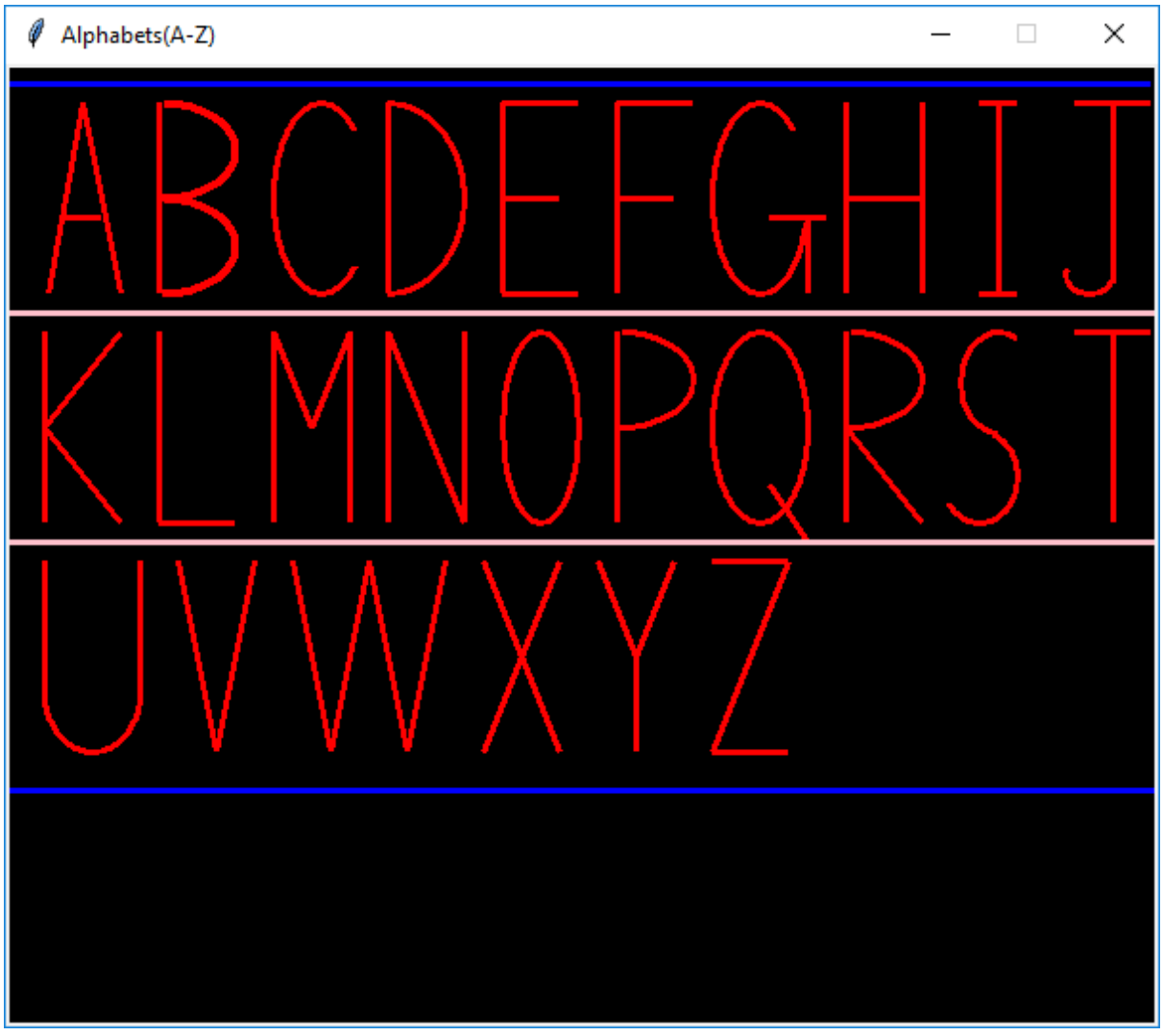
```
r2.setWidth(3)
r1.setWidth(3)
r1.draw(w)
r3.setWidth(3)
r3.draw(w)
r1.setOutline("red")
r2.setOutline("red")
r3.setOutline("red")
s1=Arc(Point(500,140),Point(540,193),60,210,"arc")
s1.draw(w)
s1.setWidth(3)
s2=Arc(Point(490,192),Point(530,240),65,-210,"arc")
s2.draw(w)
s2.setWidth(3)
s1.setOutline("red")
s2.setOutline("red")
t1=Line(Point(560,140),Point(600,140))
t1.draw(w)
t1.setWidth(3)
t2=Line(Point(580,140),Point(580,240))
t2.draw(w)
t2.setWidth(3)
t1.setOutline("red")
t2.setOutline("red")
l=Line(Point(0,250),Point(650,250))
l.draw(w)
l.setWidth(3)
l.setOutline("pink")
u1=Line(Point(20,260),Point(20,330))
u1.draw(w)
```

```
u1.setWidth(3)
u2=Line(Point(70,260),Point(70,330))
u2.draw(w)
u2.setWidth(3)
u3=Arc(Point(20,300),Point(70,360),180,180,"arc")
u3.draw(w)
u3.setWidth(3)
u1.setOutline("red")
u2.setOutline("red")
u3.setOutline("red")
v1=Line(Point(90,260),Point(110,360))
v1.draw(w)
v1.setWidth(3)
v2=Line(Point(110,360),Point(130,260))
v2.draw(w)
v2.setWidth(3)
v1.setOutline("red")
v2.setOutline("red")
w1=Line(Point(150,260),Point(170,360))
w1.draw(w)
w1.setWidth(3)
w2=Line(Point(170,360),Point(190,260))
w2.draw(w)
w2.setWidth(3)
w3=Line(Point(190,260),Point(210,360))
w3.draw(w)
w3.setWidth(3)
w4=Line(Point(210,360),Point(230,260))
w4.draw(w)
w4.setWidth(3)
```

```
w1.setOutline("red")
w2.setOutline("red")
w3.setOutline("red")
w4.setOutline("red")
x1=Line(Point(250,260),Point(290,360))
x1.draw(w)
x1.setWidth(3)
x2=Line(Point(290,260),Point(250,360))
x2.draw(w)
x2.setWidth(3)
x1.setOutline("red")
x2.setOutline("red")
y1=Line(Point(310,260),Point(330,310))
y1.draw(w)
y1.setWidth(3)
y2=Line(Point(350,260),Point(330,310))
y2.draw(w)
y2.setWidth(3)
y3=Line(Point(330,310),Point(330,360))
y3.draw(w)
y3.setWidth(3)
y1.setOutline("red")
y2.setOutline("red")
y3.setOutline("red")
z1=Line(Point(370,260),Point(410,260))
z1.draw(w)
z1.setWidth(3)
z2=Line(Point(370,360),Point(410,360))
z2.draw(w)
z2.setWidth(3)
```

```
z3=Line(Point(410,260),Point(370,360))
z3.draw(w)
z3.setWidth(3)
z1.setOutline("red")
z2.setOutline("red")
z3.setOutline("red")
```

Output:



Lower Case

Program:

```
from graphics import *
w=GraphWin("Small Alphabets(a-z",640,600)
l=Line(Point(0,10),Point(640,10))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
l=Line(Point(0,550),Point(650,550))
l.draw(w)
l.setWidth(3)
l.setOutline("red")
w.setBackground("black")
w.setBackground("black")
a1=Arc(Point(45,60),Point(95,120),40,280,"arc")
a1.draw(w)
a1.setWidth(3)
a1.setOutline("blue")
a2=Line(Point(90,60),Point(90,120))
a2.draw(w)
a2.setWidth(3)
a2.setOutline("blue")
a3=Arc(Point(90,115),Point(102,125),180,180,"arc")
a3.draw(w)
a3.setWidth(3)
a3.setOutline("blue")
b1=Line(Point(125,10),Point(125,120))
b1.draw(w)
b1.setWidth(3)
b1.setOutline("blue")
```

```
b2=Arc(Point(120,60),Point(170,120),140,-280,"arc")
b2.draw(w)
b2.setWidth(3)
b2.setOutline("blue")
c=Arc(Point(190,60),Point(240,120),50,260,"arc")
c.draw(w)
c.setWidth(3)
c.setOutline("blue")
d1=Line(Point(290,10),Point(290,120))
d1.draw(w)
d1.setWidth(3)
d1.setOutline("blue")
d2=Arc(Point(245,60),Point(295,120),40,280,"arc")
d2.draw(w)
d2.setWidth(3)
d2.setOutline("blue")
e1=Arc(Point(310,60),Point(360,120),45,300,"arc")
e1.draw(w)
e1.setWidth(3)
e1.setOutline("blue")
e2=Arc(Point(290,50),Point(350,90),10,-120,"arc")
e2.draw(w)
e2.setWidth(3)
e2.setOutline("blue")
f1=Line(Point(380,20),Point(380,120))
f1.draw(w)
f1.setWidth(3)
f1.setOutline("blue")
f2=Arc(Point(380,5),Point(410,35),0,180,"arc")
f2.draw(w)
```

```
f2.setWidth(3)
f2.setOutline("blue")
f3=Line(Point(380,60),Point(400,60))
f3.draw(w)
f3.setWidth(3)
f3.setOutline("blue")
g1=Line(Point(460,60),Point(460,140))
g1.draw(w)
g1.setWidth(3)
g1.setOutline("blue")
g2=Arc(Point(415,60),Point(465,120),40,280,"arc")
g2.draw(w)
g2.setWidth(3)
g2.setOutline("blue")
g3=Arc(Point(460,120),Point(415,160),0,-160,"arc")
g3.draw(w)
g3.setWidth(3)
g3.setOutline("blue")
h1=Line(Point(490,10),Point(490,120))
h1.draw(w)
h1.setWidth(3)
h1.setOutline("blue")
h2=Arc(Point(490,60),Point(530,100),0,180,"arc")
h2.draw(w)
h2.setWidth(3)
h2.setOutline("blue")
h3=Line(Point(530,80),Point(530,120))
h3.draw(w)
h3.setWidth(3)
h3.setOutline("blue")
```

```
i1=Line(Point(560,60),Point(560,120))
i1.draw(w)
i1.setWidth(3)
i1.setOutline("blue")
i2=Circle(Point(560,40),3)
i2.draw(w)
i2.setWidth(3)
i2.setOutline("blue")
j1=Line(Point(590,60),Point(590,150))
j1.draw(w)
j1.setWidth(3)
j1.setOutline("blue")
j2=Circle(Point(590,40),3)
j2.draw(w)
j2.setWidth(3)
j2.setOutline("blue")
j3=Arc(Point(590,130),Point(570,160),0,-90,"arc")
j3.draw(w)
j3.setWidth(3)
j3.setOutline("blue")
l1=Line(Point(0,180),Point(640,180))
l2=Line(Point(0,360),Point(640,360))
l1.draw(w)
l2.draw(w)
l1.setWidth(3)
l2.setWidth(3)
l1.setOutline("green")
l2.setOutline("green")
k1=Line(Point(50,190),Point(50,300))
```

```
k1.draw(w)
k1.setWidth(3)
k1.setOutline("blue")
k2=Line(Point(80,240),Point(50,270))
k2.draw(w)
k2.setWidth(3)
k2.setOutline("blue")
k3=Line(Point(60,260),Point(80,300))
k3.draw(w)
k3.setWidth(3)
k3.setOutline("blue")
l1=Line(Point(110,190),Point(110,300))
l1.draw(w)
l1.setWidth(3)
l1.setOutline("blue")
l2=Arc(Point(130,280),Point(110,310),180,90,"arc")
l2.draw(w)
l2.setWidth(3)
l2.setOutline("blue")
m1=Line(Point(150,240),Point(150,300))
m1.draw(w)
m1.setWidth(3)
m1.setOutline("blue")
m2=Arc(Point(150,240),Point(180,270),180,-200,"arc")
m2.draw(w)
m2.setWidth(3)
m2.setOutline("blue")
m3=Line(Point(180,260),Point(180,300))
m3.draw(w)
m3.setWidth(3)
```

```
m3.setOutline("blue")
m4=Arc(Point(180,240),Point(210,270),180,-200,"arc")
m4.draw(w)
m4.setWidth(3)
m4.setOutline("blue")
m5=Line(Point(210,250),Point(210,300))
m5.draw(w)
m5.setWidth(3)
m5.setOutline("blue")
n1=Line(Point(240,240),Point(240,300))
n1.draw(w)
n1.setWidth(3)
n1.setOutline("blue")
n2=Arc(Point(240,240),Point(270,270),180,-200,"arc")
n2.draw(w)
n2.setWidth(3)
n2.setOutline("blue")
n3=Line(Point(270,260),Point(270,300))
n3.draw(w)
n3.setWidth(3)
n3.setOutline("blue")
o=Oval(Point(290,240),Point(340,300))
o.draw(w)
o.setWidth(3)
o.setOutline("blue")
p1=Line(Point(360,240),Point(360,340))
p1.draw(w)
p1.setWidth(3)
p1.setOutline("blue")
p2=Arc(Point(355,240),Point(405,300),140,-280,"arc")
```

```
p2.draw(w)
p2.setWidth(3)
p2.setOutline("blue")
q1=Line(Point(470,240),Point(470,350))
q1.draw(w)
q1.setWidth(3)
q1.setOutline("blue")
q2=Arc(Point(425,240),Point(475,300),40,280,"arc")
q2.draw(w)
q2.setWidth(3)
q2.setOutline("blue")
q3=Line(Point(470,350),Point(500,300))
q3.draw(w)
q3.setWidth(3)
q3.setOutline("blue")
r1=Line(Point(520,240),Point(520,300))
r1.draw(w)
r1.setWidth(3)
r1.setOutline("blue")
r2=Arc(Point(520,242),Point(580,258),180,-100,"arc")
r2.draw(w)
r2.setWidth(3)
r2.setOutline("blue")
s1=Arc(Point(580,240),Point(610,270),25,260,"arc")
s1.draw(w)
s1.setWidth(3)
s1.setOutline("blue")
s2=Arc(Point(580,270),Point(610,300),105,-270,"arc")
s2.draw(w)
s2.setWidth(3)
```

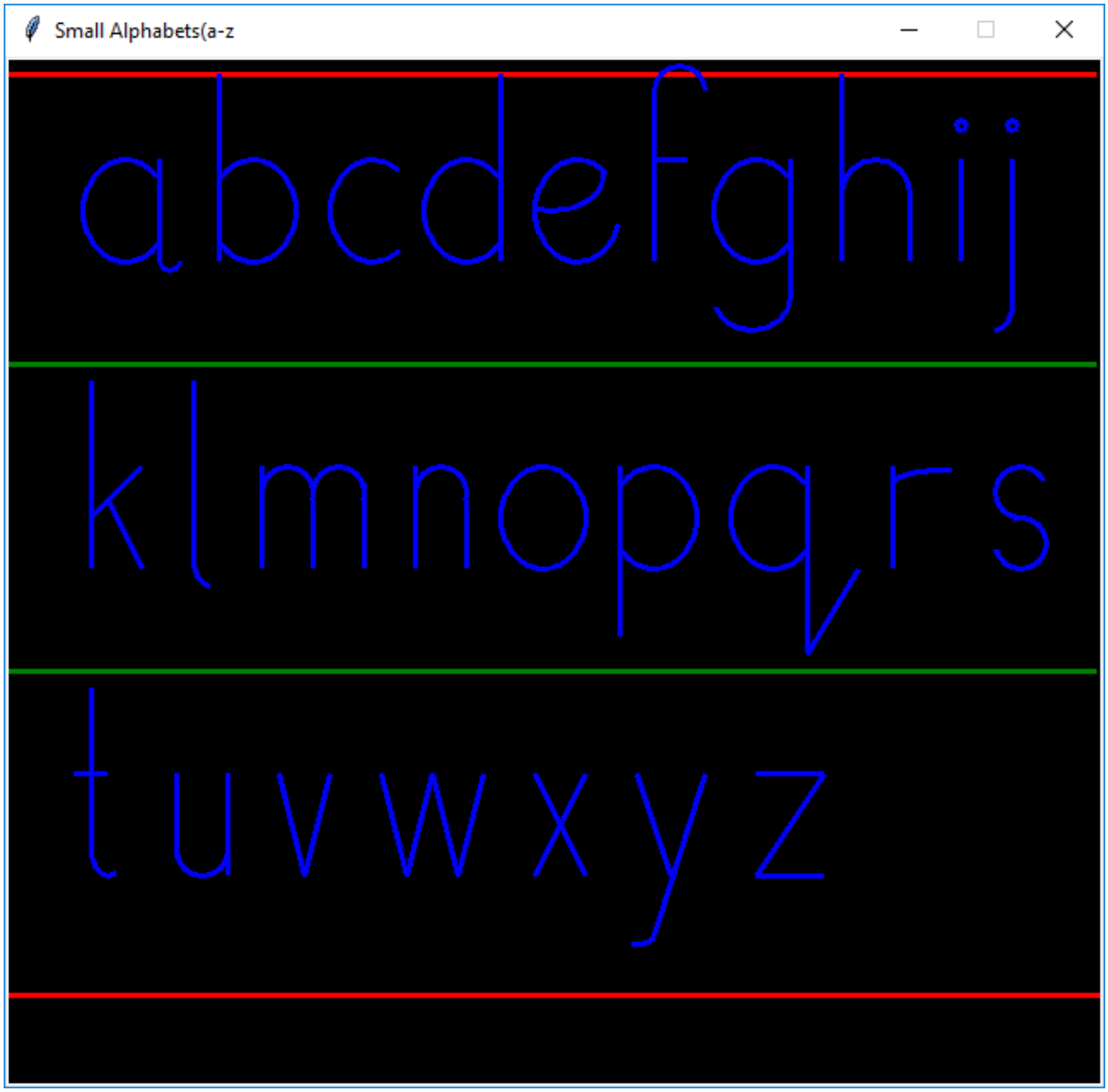
```
s2.setOutline("blue")
t1=Line(Point(50,370),Point(50,470))
t1.draw(w)
t1.setWidth(3)
t1.setOutline("blue")
t2=Arc(Point(50,450),Point(70,480),180,120,"arc")
t2.draw(w)
t2.setWidth(3)
t2.setOutline("blue")
t3=Line(Point(40,420),Point(60,420))
t3.draw(w)
t3.setWidth(3)
t3.setOutline("blue")
u1=Line(Point(100,420),Point(100,470))
u1.draw(w)
u1.setWidth(3)
u1.setOutline("blue")
u2=Arc(Point(100,450),Point(130,480),170,200,"arc")
u2.draw(w)
u2.setWidth(3)
u2.setOutline("blue")
u3=Line(Point(130,420),Point(130,480))
u3.draw(w)
u3.setWidth(3)
u3.setOutline("blue")
v1=Line(Point(160,420),Point(175,480))
v1.draw(w)
v1.setWidth(3)
v1.setOutline("blue")
```

```
v2=Line(Point(190,420),Point(175,480))
v2.draw(w)
v2.setWidth(3)
v2.setOutline("blue")
w1=Line(Point(220,420),Point(235,480))
w1.draw(w)
w1.setWidth(3)
w1.setOutline("blue")
w2=Line(Point(250,420),Point(235,480))
w2.draw(w)
w2.setWidth(3)
w2.setOutline("blue")
w3=Line(Point(250,420),Point(265,480))
w3.draw(w)
w3.setWidth(3)
w3.setOutline("blue")
w4=Line(Point(280,420),Point(265,480))
w4.draw(w)
w4.setWidth(3)
w4.setOutline("blue")
x1=Line(Point(310,420),Point(340,480))
x1.draw(w)
x1.setWidth(3)
x1.setOutline("blue")
x2=Line(Point(340,420),Point(310,480))
x2.draw(w)
x2.setWidth(3)
x2.setOutline("blue")
y1=Line(Point(370,420),Point(390,480))
y1.draw(w)
```

```
y1.setWidth(3)
y1.setOutline("blue")
y2=Line(Point(410,420),Point(380,515))
y2.draw(w)
y2.setWidth(3)
y2.setOutline("blue")
y3=Arc(Point(360,510),Point(380,520),0,-110,"arc")
y3.draw(w)
y3.setWidth(3)
y3.setOutline("blue")
z1=Line(Point(440,420),Point(480,420))
z1.draw(w)
z1.setWidth(3)
z1.setOutline("blue")
z2=Line(Point(480,420),Point(440,480))
z2.draw(w)
z2.setWidth(3)
z2.setOutline("blue")
z3=Line(Point(440,480),Point(480,480))
z3.draw(w)
z3.setWidth(3)
z3.setOutline("blue")
```

Roll no: 37
Name: jaimini patel

Output:



Ex no: 4

Date: 4-12-2019

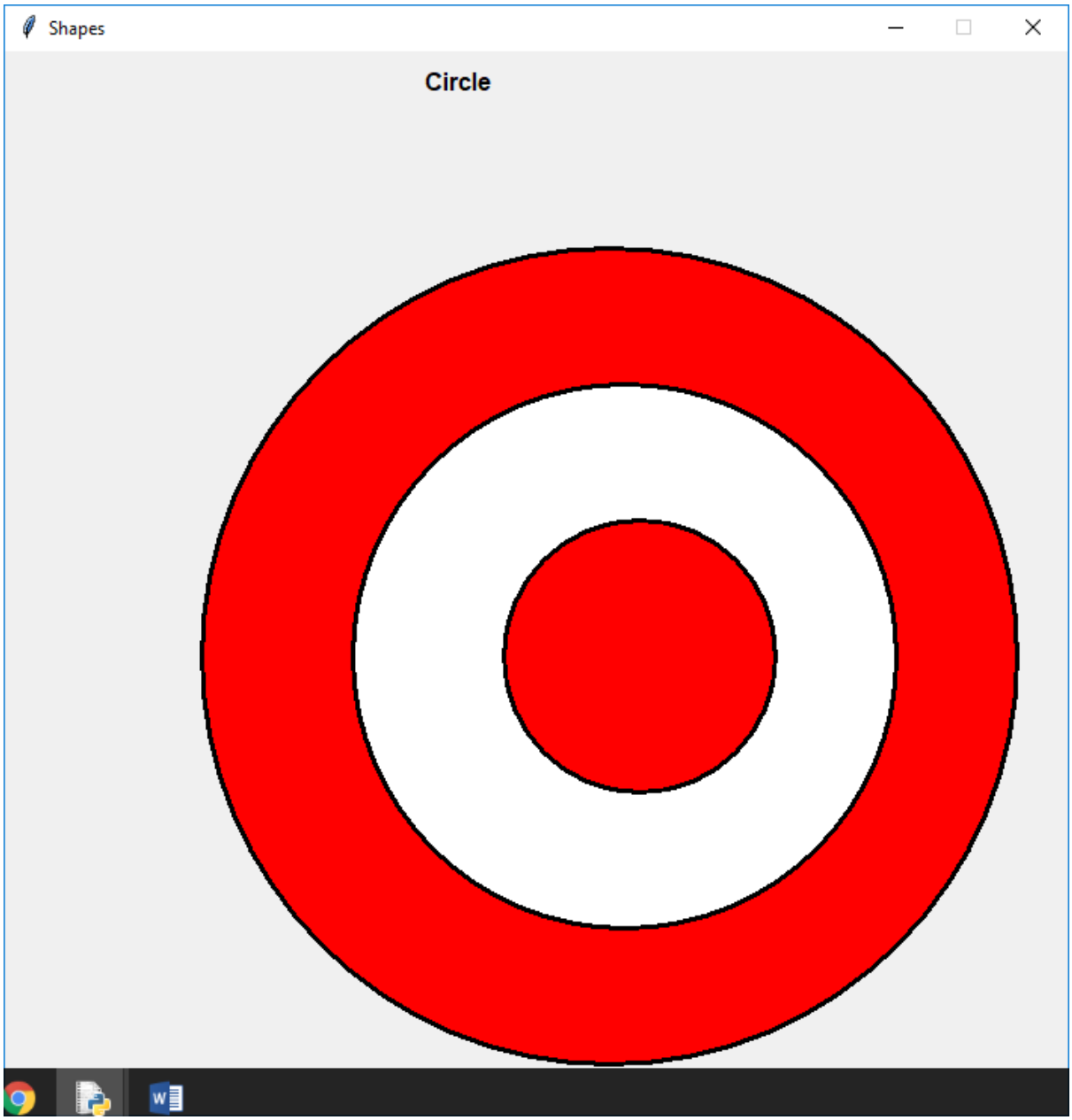
Problem Statement:

Draw the Target symbol (a set of concentric circles, alternating red and white) in a graphics window that is 200 pixels wide by 200 pixels high. Hint: Draw the largest circle first in red, then draw the next smaller circle in white, then draw the next smaller circle in red.

Program:

```
from graphics import *
w=GraphWin("Shapes",700,700)
label=Text(Point(300,20),"Circle")
label.draw(w)
label.setStyle("bold")
center=Point(400,400)
circ=Circle(center,270)
circ.setFill("red")
circ.setWidth(3)
circ.draw(w)
center=Point(410,400)
circ=Circle(center,180)
circ.setFill("white")
circ.setWidth(3)
circ.draw(w)
center=Point(420,400)
circ=Circle(center,90)
circ.setFill("red")
circ.setWidth(3)
circ.draw(w)
```

Output:



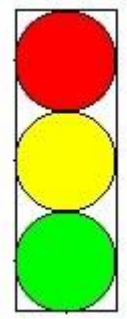
Roll no: 37
Name: jaimini patel

Ex no: 5

Date: 4-12-2019

Problem Statement:

Draw a simple traffic light in a graphics window that is 200 pixels wide by 200 pixels high. The three lights should have a diameter of 50 pixels each, and the traffic light should be centered in the graphics window.

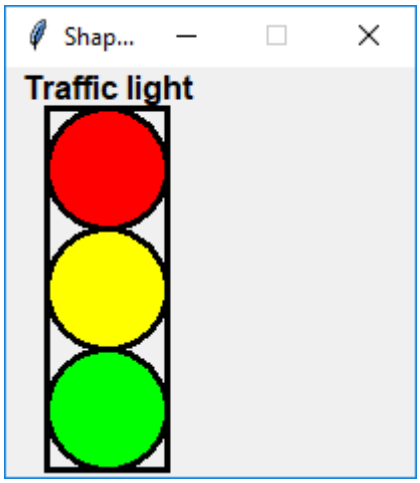


Program:

```
from graphics import *
w=GraphWin("Shapes",200,200)
label=Text(Point(50,10),"Traffic light")
label.draw(w)
label.setStyle("bold")
rect=Rectangle(Point(20,20),Point(80,200))
rect.draw(w)
rect.setWidth(3)
center=Point(50,50)
circ=Circle(center,30)
circ.setFill("red")
circ.setWidth(3)
circ.draw(w)
center=Point(50,110)
circ=Circle(center,30)
```

```
circ.setFill("yellow")
circ.setWidth(3)
circ.draw(w)
center=Point(50,170)
circ=Circle(center,30)
circ.setFill(color_rgb(0,255,0))
circ.setWidth(3)
circ.draw(w)
```

Output:



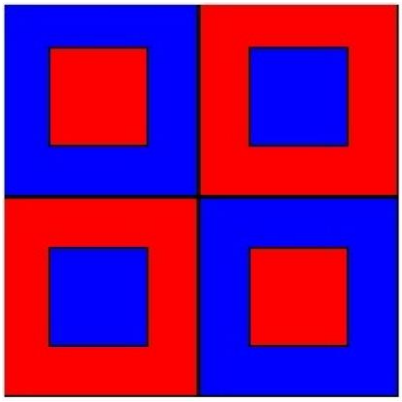
Roll no: 37
Name: jaimini patel

Ex no: 6

Date: 4-12-2019

Problem Statement:

Create the image shown below.



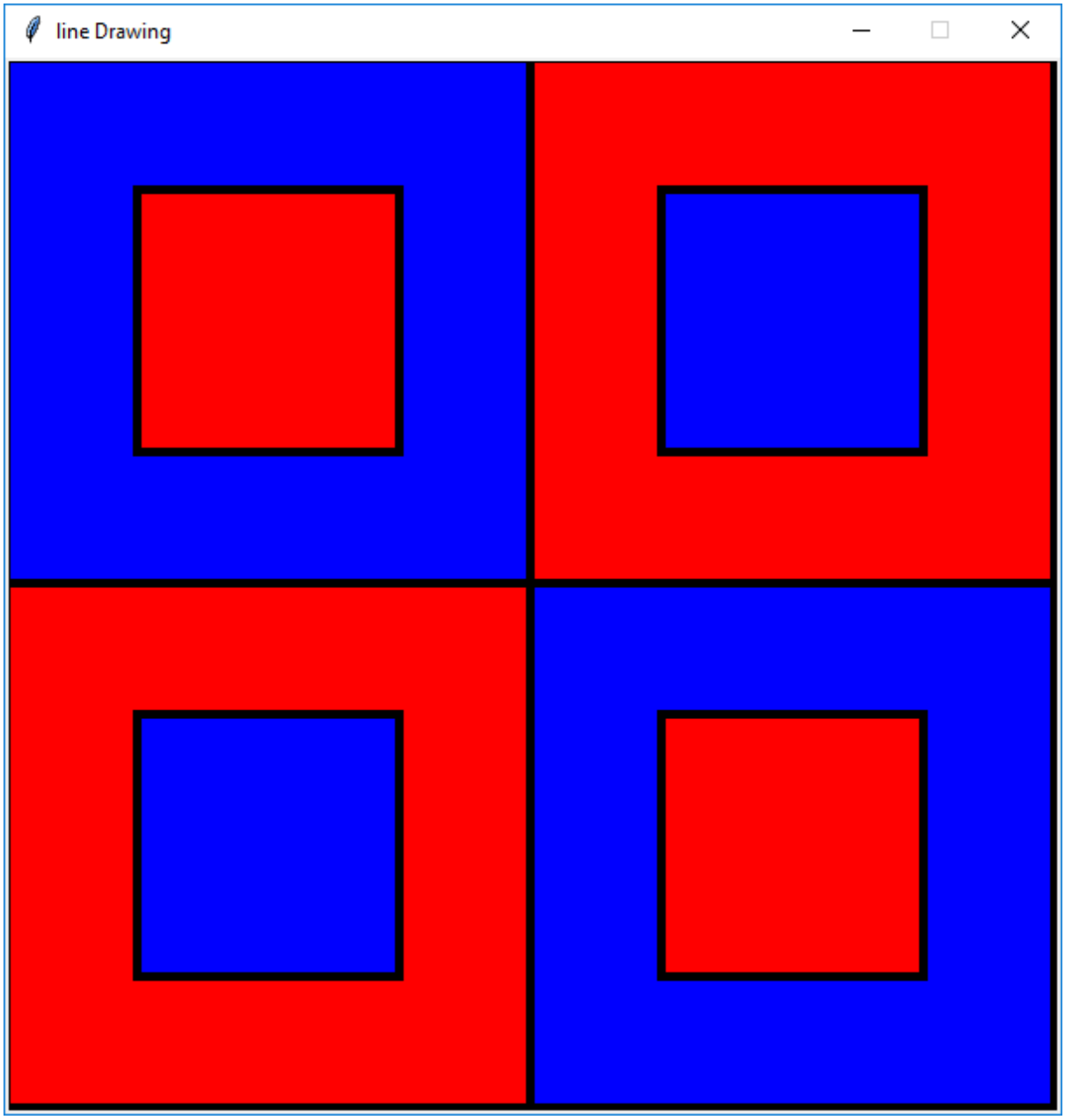
Program:

```
from graphics import *
w=GraphWin("line Drawing",600,600)
rect=Rectangle(Point(0,0),Point(300,300))
rect.setWidth(5)
rect.draw(w)
rect.setFill("blue")
rect=Rectangle(Point(75,75),Point(225,225))
rect.setWidth(5)
rect.draw(w)
rect.setFill("red")
rect=Rectangle(Point(300,0),Point(600,300))
rect.setWidth(5)
rect.draw(w)
rect.setFill("red")
rect=Rectangle(Point(375,75),Point(525,225))
rect.setWidth(5)
```

```
rect.draw(w)
rect.setFill("blue")
rect=Rectangle(Point(0,300),Point(300,600))
rect.setWidth(5)
rect.draw(w)
rect.setFill("red")
rect=Rectangle(Point(75,375),Point(225,525))
rect.setWidth(5)
rect.draw(w)
rect.setFill("blue")
rect=Rectangle(Point(300,300),Point(600,600))
rect.setWidth(5)
rect.draw(w)
rect.setFill("blue")
rect=Rectangle(Point(375,375),Point(525,525))
rect.setWidth(5)
rect.draw(w)
rect.setFill("red")
```

Roll no: 37
Name: jaimini patel

Output:



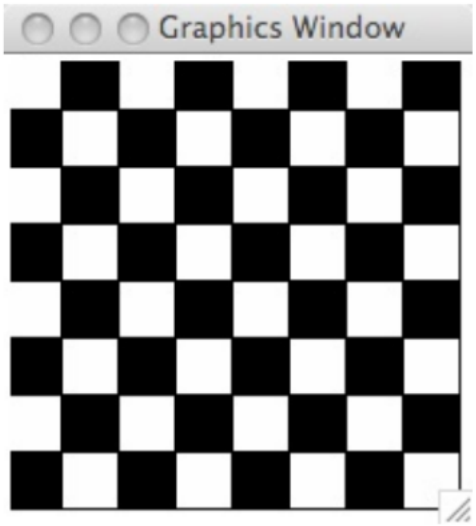
Roll no: 37
Name: jaimini patel

Ex no: 7

Date: 12-12-2019

Problem Statement:

Create a checkerboard of white and black squares in a graphics window that is 200 pixels wide by 200 pixels high. Each square should be 25 X 25. Can you simplify this program using loops?

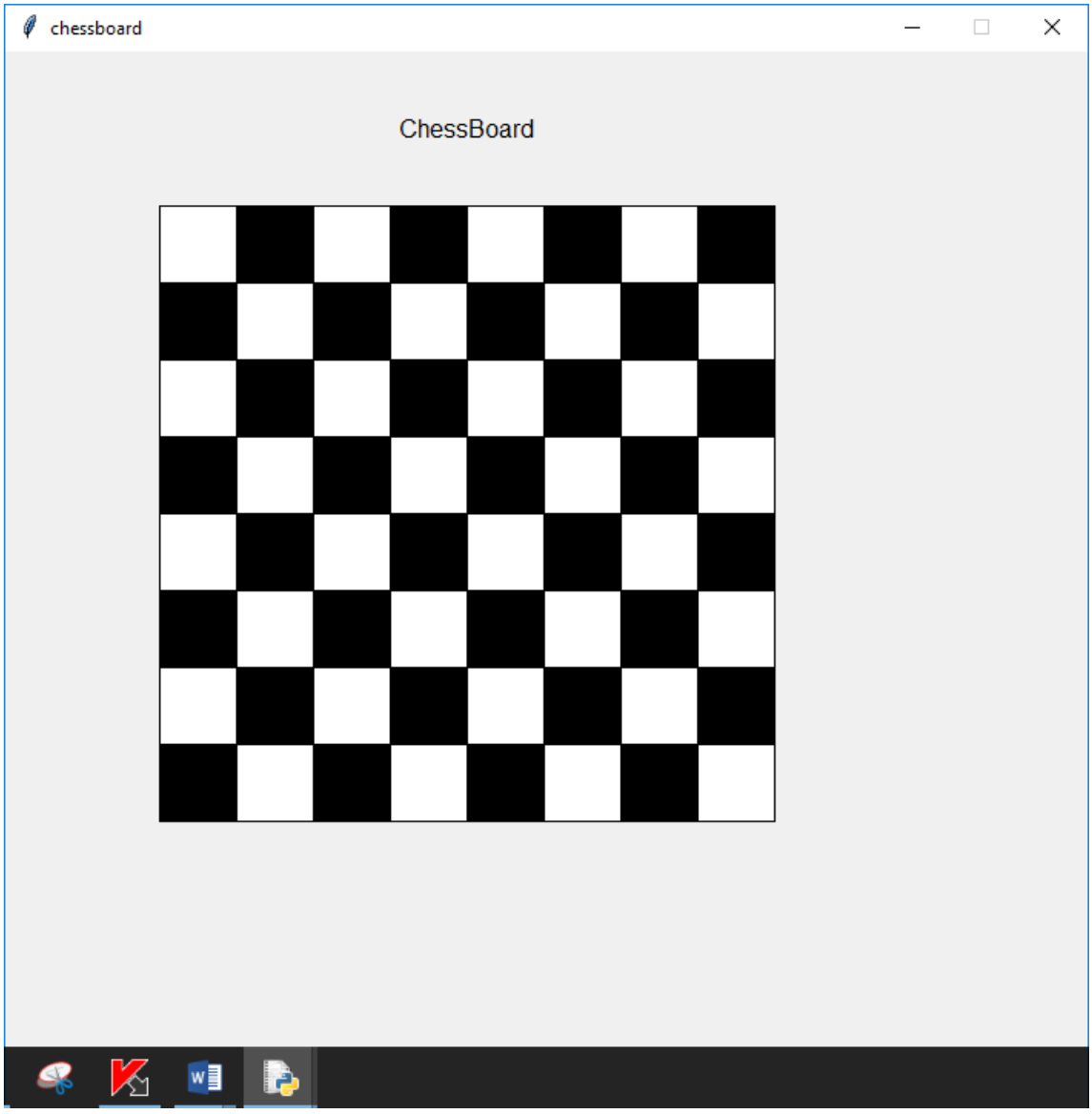


Program:

```
from graphics import *
w=GraphWin("chessboard",700,700)
label=Text(Point(300,50),"ChessBoard")
label.draw(w)
a="black"
b="white"
x=100
y=100
for j in range(8):
    for i in range(8):
        l=Rectangle(Point(x,y),Point(x+50,y+50))
        l.draw(w)
```

```
        if (i+j)%2!=0:
            l.setFill(a)
        else:
            l.setFill(b)
        x=x+50
    y=y+50
    x=100
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 8

Date: 12-12-2019

Problem Statement:

Write a Python program to show Snowman using basic shapes

Program:

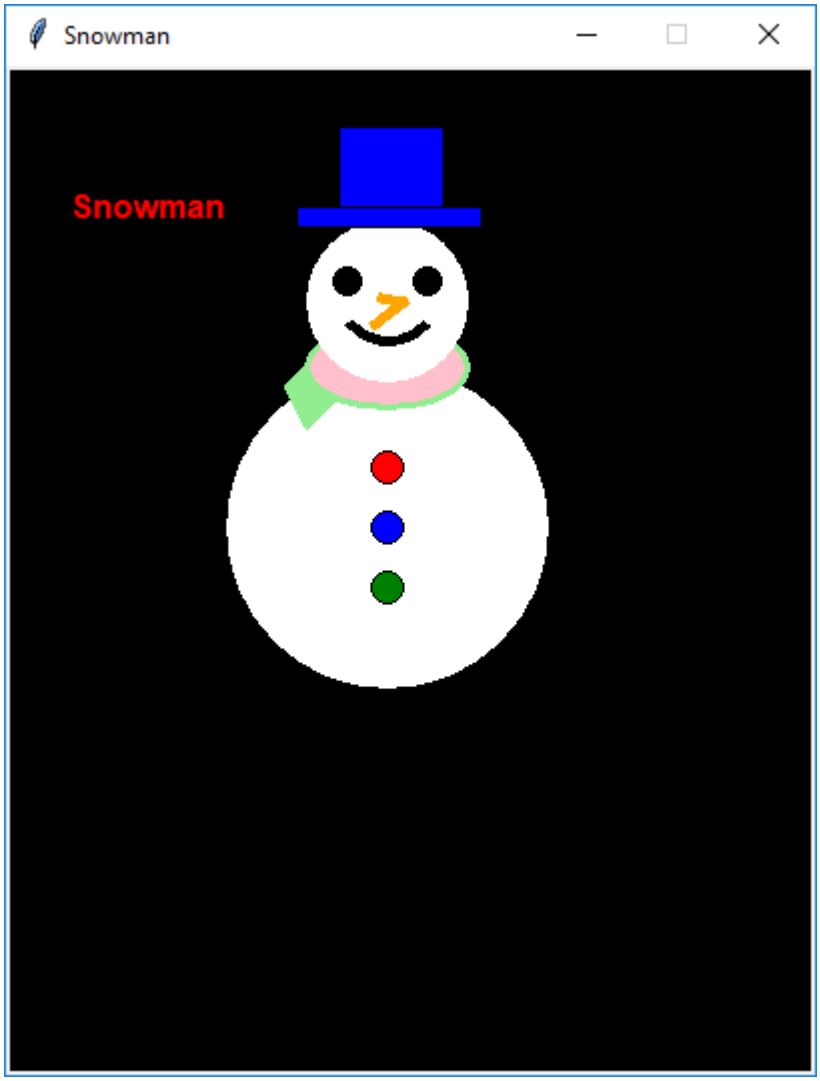
```
from graphics import*
x=400
y=500
win=GraphWin("Snowman",x,y)
label=Text(Point(70,70),"Snowman")
label.draw(win)
label.setStyle("bold")
label.setTextColor("red")
rect=Rectangle(Point(166,30),Point(218,70))
rect.draw(win)
rect.setFill("blue")
#big circle
center=Point(190,230)
circ=Circle(center,80)
circ.draw(win)
circ.setFill("white")
circ.setOutline("white")
#scraf
r=Polygon(Point(150,150),Point(140,160),
        Point(150,180),Point(200,130))
#         Point(180,150),Point(190,160))
r.setFill("lightgreen")
r.setWidth(3)
```

```
r.setOutline("lightgreen")
r.draw(win)
r=Oval(Point(150,170),Point(230,130))
r.draw(win)
r.setFill("pink")
r.setWidth(3)
r.setOutline("lightgreen")
#small circle
center=Point(190,117)
circ=Circle(center,40)
circ.draw(win)
circ.setOutline("white")
circ.setFill("white")
rect=Rectangle(Point(145,70),Point(237,80))
rect.draw(win)
rect.setFill("blue")
#left eye
center=Point(170,107)
circ=Circle(center,7)
circ.setFill('black')
circ.draw(win)
#right eye
center=Point(210,107)
circ=Circle(center,7)
circ.setFill('black')
circ.draw(win)
#nose
l=Line(Point(185,115),Point(200,118))
l.draw(win)
l.setFill("orange")
```

```
l.setWidth(5)
l=Line(Point(200,116),Point(182,130))
l.draw(win)
l.setFill("orange")
l.setWidth(5)
#smile
r=Arc(Point(160,57),Point(221,137),230,80,"arc")
r.draw(win)
r.setWidth(5)
#buttons
center=Point(190,200)
circ=Circle(center,8)
circ.setFill('red')
circ.draw(win)
center=Point(190,230)
circ=Circle(center,8)
circ.setFill('blue')
circ.draw(win)
center=Point(190,260)
circ=Circle(center,8)
circ.setFill('green')
circ.draw(win)
win.setBackground('black')
```

Roll no: 37
Name: jaimini patel

Output:



Roll no: 37
Name: jaimini patel

Ex no: 9

Date: 12-12-2019

Problem Statement:

Write a Python program to show House using basic shapes

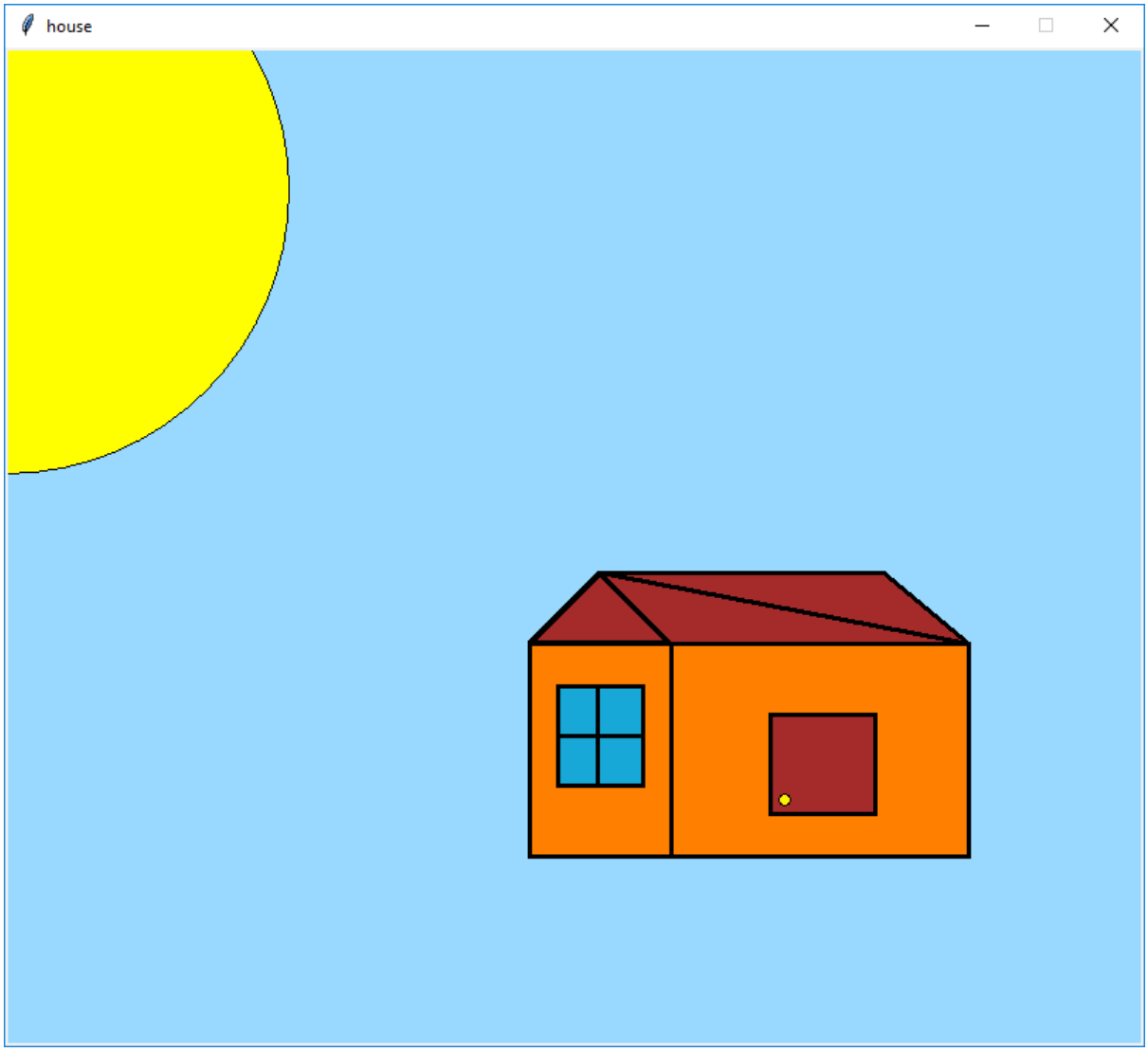
Program:

```
from graphics import *
w=GraphWin("house",800,700)
triangle=Polygon(Point(370,420),Point(420,370),Point(470,420))
triangle.setFill('brown')
triangle.setWidth(4)
triangle.draw(w)
w.setBackground(color_rgb(153,217,255))
center=Point(0,100)
circ=Circle(center,200)
circ.setFill('yellow')
circ.draw(w)
rect=Rectangle(Point(370,570),Point(470,420))
rect.setFill(color_rgb(255,128,0))
rect.draw(w)
rect.setWidth(3)
rect=Rectangle(Point(470,570),Point(680,420))
rect.setFill(color_rgb(255,128,0))
rect.setWidth(3)
rect.draw(w)
rect=Rectangle(Point(540,540),Point(614,470))
rect.setFill('brown')
rect.draw(w)
rect.setWidth(3)
```

```
rect=Rectangle(Point(390,520),Point(450,450))
rect.setFill(color_rgb(24,168,216))
rect.draw(w)
rect.setWidth(3)
line=Line(Point(418,520),Point(418,450))
line.setWidth(3)
line.draw(w)
line=Line(Point(390,485),Point(450,485))
line.setWidth(3)
line.draw(w)
t=Polygon(Point(420,370),Point(620,370),Point(680,420))
t.setFill('brown')
t.setWidth(3)
t.draw(w)
t=Polygon(Point(420,370),Point(680,420),Point(470,420))
t.setFill('brown')
t.setWidth(3)
t.draw(w)
center=Point(550,530)
circ=Circle(center,4)
circ.setFill(color_rgb(255,242,0))
circ.draw(w)
```

Roll no: 37
Name: jaimini patel

Output:



Roll no: 37
Name: jaimini patel

Ex no: 10

Date: 12-12-2020

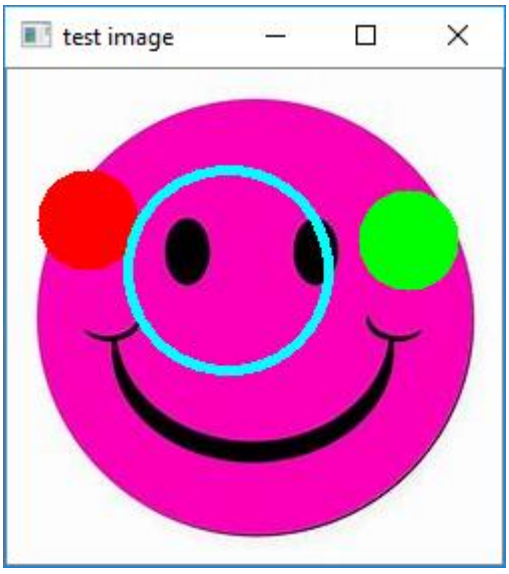
Problem Statement:

Colour handling in opencv

Program:

```
import cv2
image=cv2.imread('face.jpg')
cv2.imshow('original',image)
cv2.circle(image,(200,85),25,(0,255,0),-1)
cv2.circle(image,(40,75),25,(0,0,255),-1)
cv2.circle(image,(110,100),50,(255,255,0),3)
cv2.imshow("test image",image)
cv2.waitKey(0)
cv2.destoryAllwindow()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 11

Date: 02-01-2020

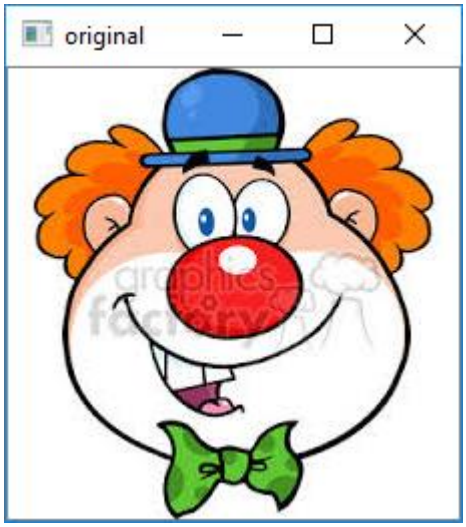
Problem Statement:

Negative of image using OpenCv.

Program:

```
import cv2
image=cv2.imread('clown1.jpg')
cv2.imshow('original',image)
img_not=cv2.bitwise_not(image)
cv2.imshow("Negative",img_not)
cv2.waitKey(0)
cv2.destoryAllWindows()
```

Output:



Ex no: 12

Date: 02-01-2020

Problem Statement:

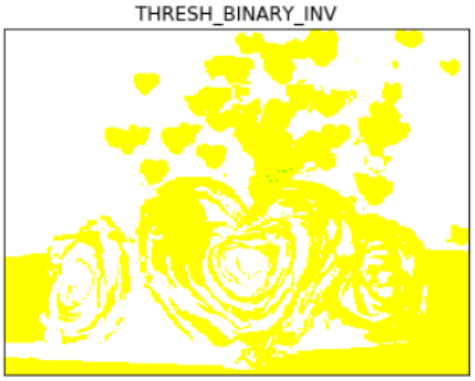
Thresholding using OpenCv.

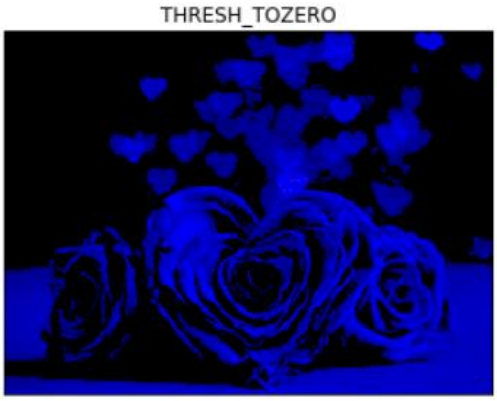
Program:

```
from cv2 import *
from matplotlib import pyplot as plt
img1=cv2.imread('download.jpg')
img=cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)
ret,thresh1=threshold(img1,120,255,THRESH_BINARY)
ret,thresh2=threshold(img1,120,255,THRESH_BINARY_INV)
ret,thresh3=threshold(img1,120,255,THRESH_TRUNC)
ret,thresh4=threshold(img1,120,255,THRESH_TOZERO)
ret,thresh5=threshold(img1,120,255,THRESH_TOZERO_INV)
images=[img,thresh1,thresh2,thresh3,thresh4,thresh5]
titles=["Original","THRESH_BINARY","THRESH_BINARY_INV","THRESH_TRUNC","
THRESH_TOZERO","THRESH_TOZERO_INV"]
for i in range(6):
    plt.subplot()
    plt.imshow(images[i],cmap="gray")
    plt.title(titles[i])
    plt.xticks([])
    plt.yticks([])
    plt.show()
waitKey(0)
destroyAllWindows()
```

Roll no: 37
Name: jaimini patel

Output:





Roll no: 37
Name: jaimini patel

Ex no: 13

Date: 02-01-2020

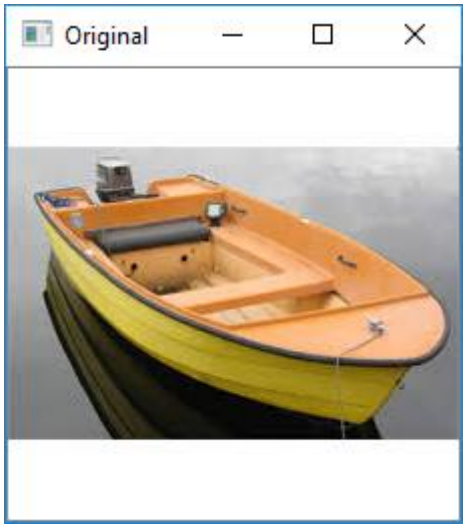
Problem Statement:

Power law transformation using OpenCv.

Program:

```
import cv2
import numpy as np
img=cv2.imread('boat.jpg')
cv2.imshow('Original',img)
im1=img/255.0
im_power_law_transformation=cv2.pow(im1,0.6)
cv2.imshow("power law tansformation",im_power_law_transformation)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 14

Date: 09-01-2020

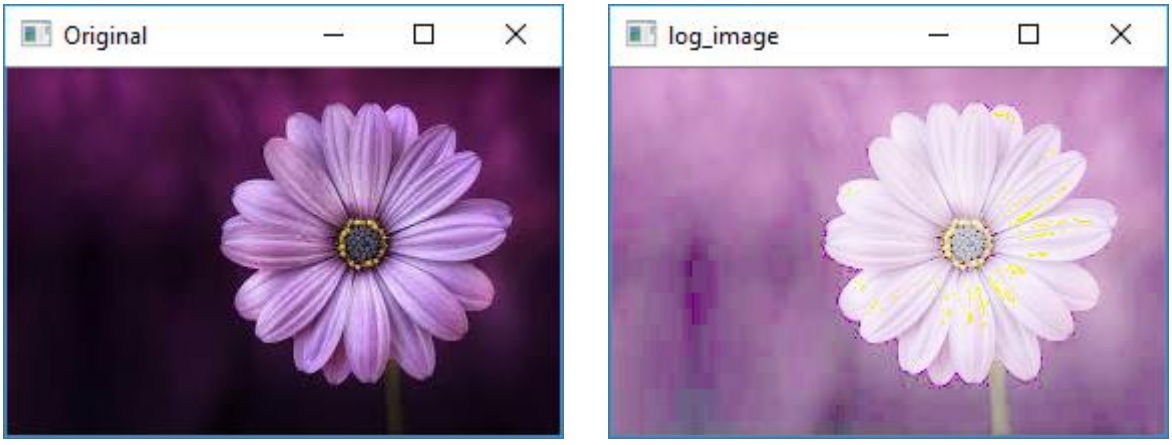
Problem Statement:

Log Transformation using OpenCv

Program:

```
import cv2
import numpy as np
img=cv2.imread('flower.jpg')
img_log=(np.log(img+1)/(np.log(1+np.max(img))))*255
img_log=np.array(img_log,dtype=np.uint8)
cv2.imshow('log_image',img_log)
cv2.imshow('Original',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Ex no: 15

Date: 09-01-2020

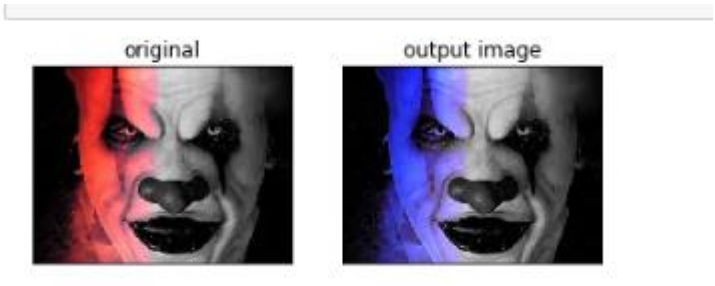
Problem Statement:

Contrast Stretching using OpenCv

Program:

```
import cv2
from matplotlib import pyplot as plt
img1=cv2.imread('images.jpg')
img=cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)
nmax=255
nmin=0
out=cv2.normalize(img1,None,alpha=nmin,beta=nmax,norm_type=cv2.NORM_MINMAX)
plt.subplot(1,2,1),plt.imshow(img)
plt.title('original'),plt.xticks([]),plt.yticks([])
plt.subplot(1,2,2),plt.imshow(out,cmap='gray')
plt.title('output image'),plt.xticks([]),plt.yticks([])
plt.show()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 16

Date: 09-01-2020

Problem Statement:

Brightness using OpenCv

Program:

```
from cv2 import *
from matplotlib import pyplot as plt
x=30
img=cv2.imread('flower.jpg')
img1=cv2.imread('flower.jpg')
img=cv2.cvtColor(img,cv2.COLOR_BGR2RGB)
m=len(img)
n=len(img[0])
bright=img1
for i in range(1,m,1):
   for j in range(1,n,1):
      bright[i,j]=img[i,j]+x
plt.subplot(1,2,1)
plt.imshow(img,cmap='gray')
plt.title('Original Image')
plt.xticks([])
plt.yticks([])
plt.subplot(1,2,2)
plt.imshow(bright,cmap='gray')
plt.title('Brighter Image')
plt.xticks([])
plt.yticks([])
plt.show()
```

Output:

Ex no: 17

Date: 09-01-2020

Problem Statement:

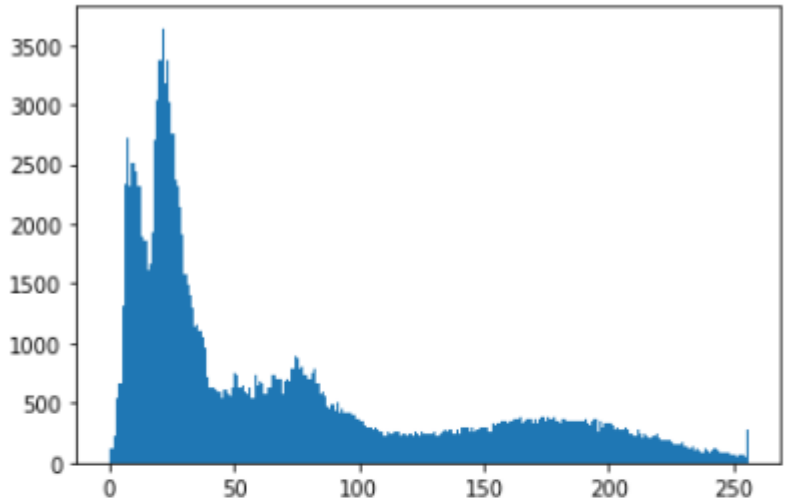
Histogram Processing using OpenCv.

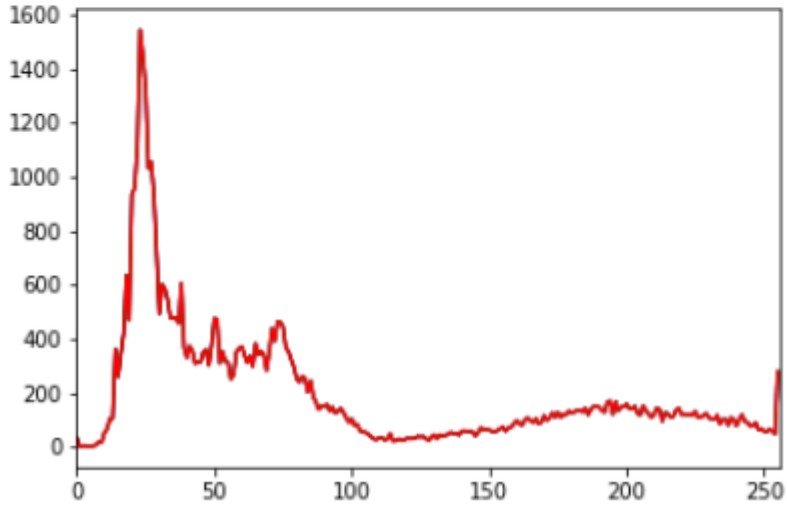
Program:

```
import cv2
import numpy as np
from matplotlibimport pyplot as plt
%matplotlib inline
img=cv2.imread('flower.jpg')
cv2.imshow("original",img)
histogram=cv2.calcHist([img],[0],None,[256],[0,256])
plt.hist(img.ravel(),256,[0,256]);plt.show();
color=('b','g','r')
for _i,col in enumerate(color):
    histogram1=cv2.calcHist([img],[i],None,[256],[0,256])
    plt.plot(histogram1,color=col)
    plt.xlim([0,256])
plt.show()
cv2.waitKey(0)
cv2.distoryAllWindows()
```

Roll no: 37
Name: jaimini patel

Output:





Roll no: 37
Name: jaimini patel

Ex no: 18

Date: 16-01-2020

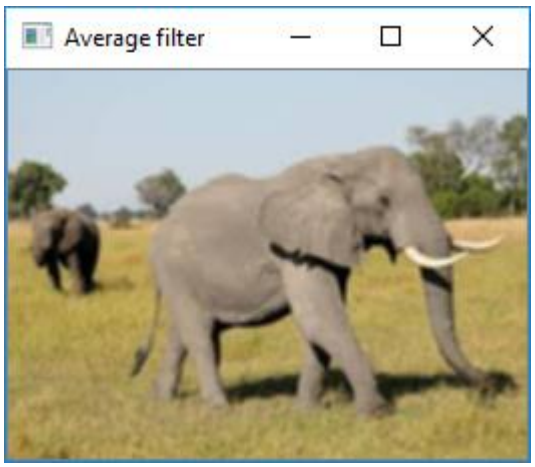
Problem Statement:

Average Filter using OpenCv

Program:

```
import cv2
import numpy as np
image=cv2.imread('elephant.jpg')
cv2.imshow('original image',image)
avg=cv2.blur(image,(3,3))
cv2.imshow('Average filter',avg)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 19

Date: 16-01-2020

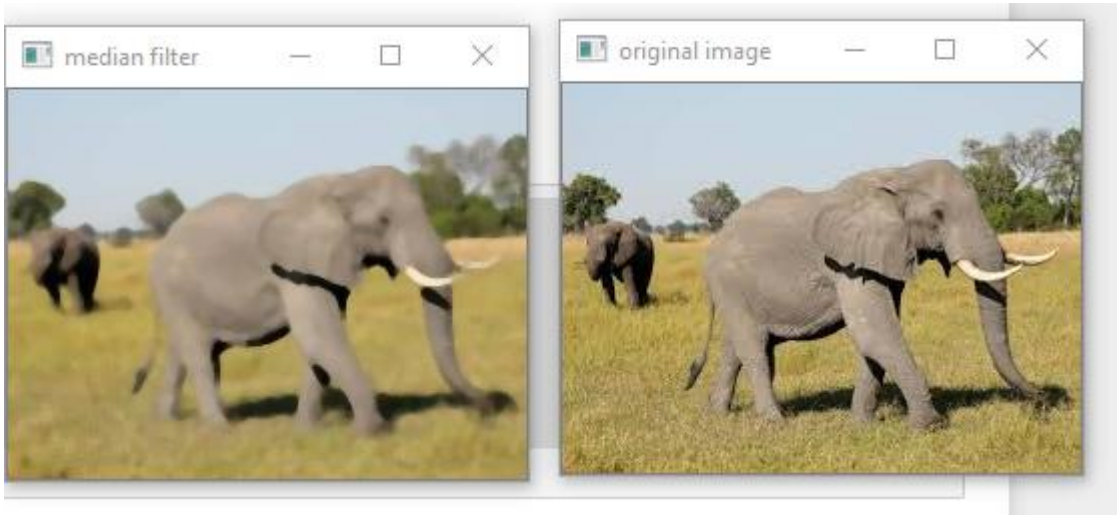
Problem Statement:

Median Filter using OpenCv

Program:

```
import cv2
import numpy as np
image=cv2.imread('elephant.jpg')
cv2.imshow('original image',image)
medi=cv2.medianBlur(image,5)
cv2.imshow('median filter',medi)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 20

Date: 16-01-2020

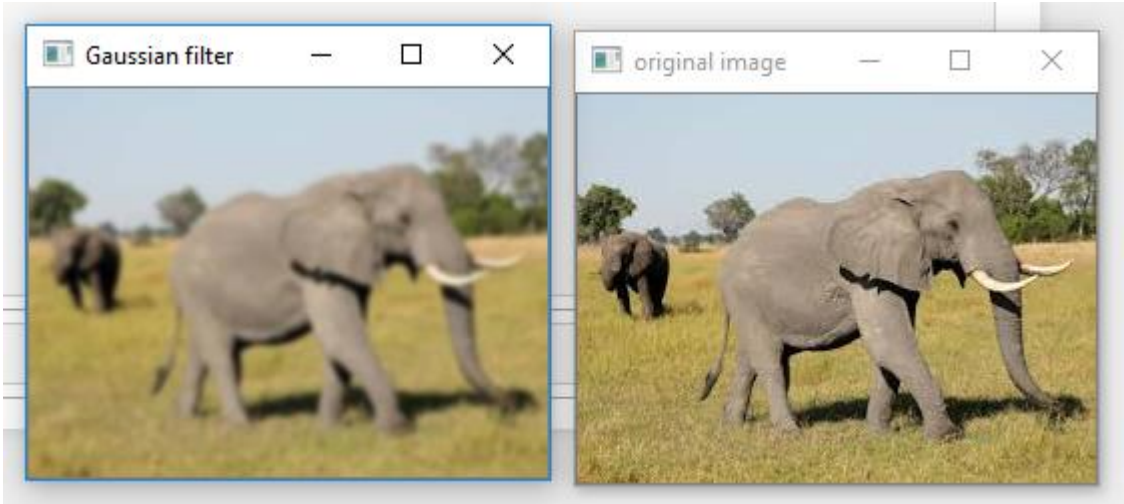
Problem Statement:

Gaussian Blur using OpenCv

Program:

```
import cv2
import numpy as np
image=cv2.imread('elephant.jpg')
cv2.imshow('original image',image)
gauss=cv2.GaussianBlur(image,(7,7),0)
cv2.imshow('Gaussian filter',gauss)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 21

Date: 16-01-2020

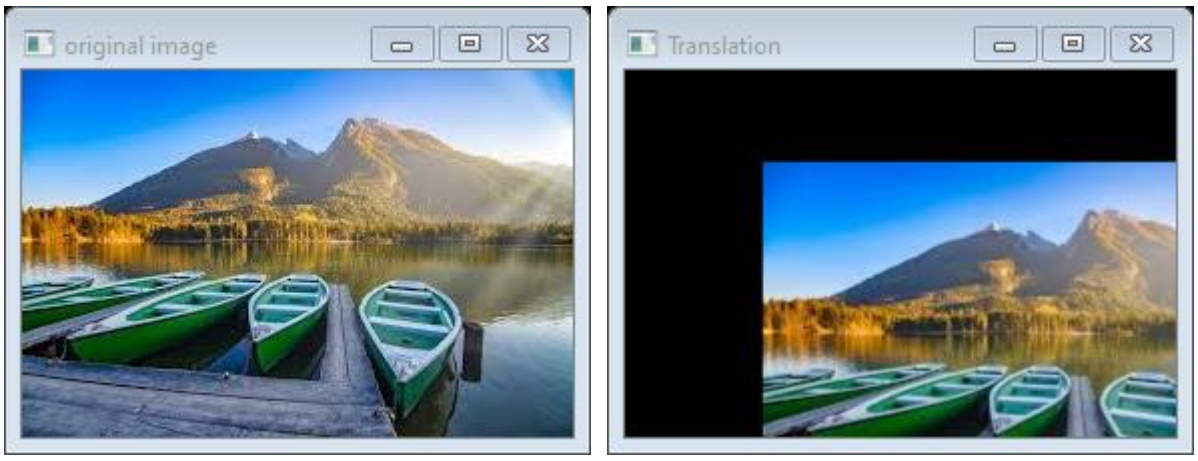
Problem Statement:

Translation of image using OpenCv.

Program:

```
import cv2
import numpy as np
image=cv2.imread("translation.jpg")
height ,width=image.shape[:2]
quarter_height,quarter_width=height/4,width/4
T=np.float32([[1,0,quarter_width],[0,1,quarter_height]])
img_translation=cv2.warpAffine(image,T,(width,height))
cv2.imshow("original image",image)
cv2.imshow("Translation",img_translation)
cv2.waitKey()
cv2.destoryAllwindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 22

Date: 28-01-2020

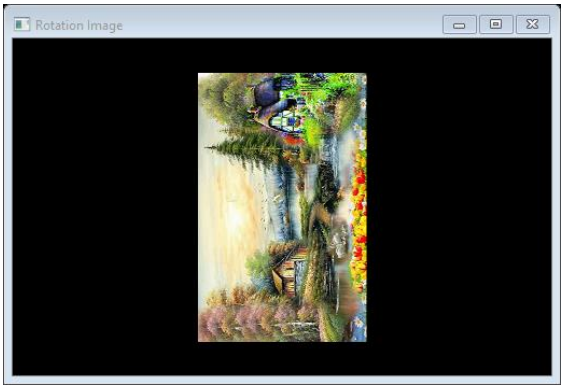
Problem Statement:

Rotation of image using OpenCv.

Program:

```
import cv2
import numpy as np
image=cv2.imread("rotation.jpg")
height ,width=image.shape[0:2]
rotationMatrix=cv2.getRotationMatrix2D((width/2,height/2),90,.5)
rotationImage=cv2.warpAffine(image,rotationMatrix,(width,height))
cv2.imshow("original image",image)
cv2.imshow("Rotation Image",rotationImage)
cv2.waitKey(0)
cv2.destoryAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 23

Date: 28-01-2020

Problem Statement:

Cropping of image using OpenCv.

Program:

```
import cv2
import numpy as np
image=cv2.imread("cropping.jpg")
height,width=image.shape[0:2]
startRow=int(height*.5)
startCol=int(width*.5)
endRow=int(height*.75)
endCol=int(width*.75)
croppingImage=image[startRow:endRow,startCol:endCol]
cv2.imshow("original image",image)
cv2.imshow("Cropping image",croppingImage)
cv2.waitKey(0)
cv2.destoryAllWindows()
```

Roll no: 37
Name: jaimini patel

Output:

Ex no: 24

Date: 28-01-2020

Problem Statement:

Sharpening Filter using OpenCv.

Program:

```
import cv2
import numpy as np
#reading in and displaying our image
image=cv2.imread('sharp.jpg')
cv2.imshow('Original',image)
#create our sharpening kernel,it must equal to one eventually
kernel_sharpening=np.array([[-1,-1,-1],[-1,9,-1],[-1,-1,-1]])
sharpened=cv2.filter2D(image,-1,kernel_sharpening)
cv2.imshow('Image Sharpening',sharpened)
cv2.waitKey(0)
```

Output:

Roll no: 37
Name: jaimini patel

Ex no: 25

Date: 28-01-2020

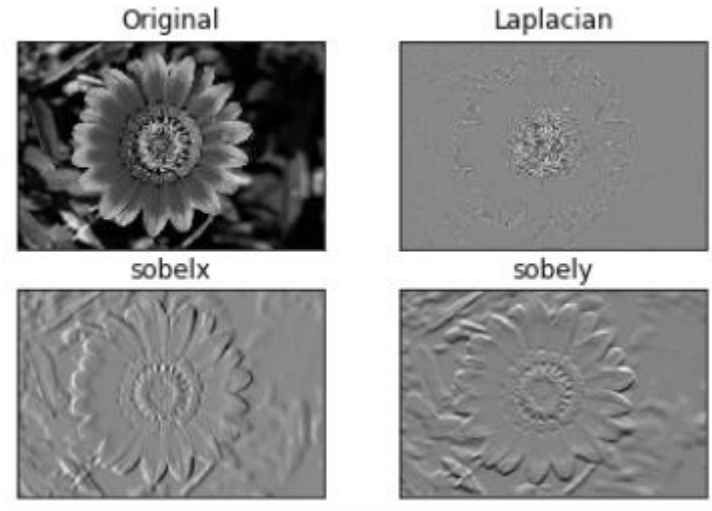
Problem Statement:

Sobel and Laplacian Filter using OpenCv

Program:

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
img=cv2.imread('sharp.jpg',0)
laplacian=cv2.Laplacian(img,cv2.CV_64F)
sobelx=cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
sobely=cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
plt.subplot(2,2,1),plt.imshow(img,cmap='gray')
plt.title('Original'),plt.xticks([]),plt.yticks([])
plt.subplot(2,2,2),plt.imshow(laplacian,cmap='gray')
plt.title('Laplacian'),plt.xticks([]),plt.yticks([])
plt.subplot(2,2,3),plt.imshow(sobelx,cmap='gray')
plt.title('sobelx'),plt.xticks([]),plt.yticks([])
plt.subplot(2,2,4),plt.imshow(sobely,cmap='gray')
plt.title('sobely'),plt.xticks([]),plt.yticks([])
plt.show()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 26

Date: 05-02-2020

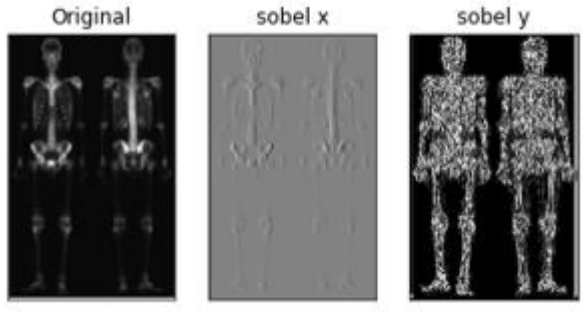
Problem Statement:

Absolute sharpening using OpenCv

Program:

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
img=cv2.imread('Capture.jpg',0)
sobelx=cv2.Sobel(img,cv2.CV_64F,1,0,ksize=5)
sobely=cv2.Sobel(img,cv2.CV_64F,0,1,ksize=5)
abs_sobelx64f=np.absolute(sobelx)
sobelx_8u=np.uint8(abs_sobelx64f)
plt.subplot(1,3,1),plt.imshow(img,cmap='gray')
plt.title('Original'),plt.xticks([]),plt.yticks([])
plt.subplot(1,3,2),plt.imshow(sobelx,cmap='gray')
plt.title('sobel x'),plt.xticks([]),plt.yticks([])
plt.subplot(1,3,3),plt.imshow(sobelx_8u,cmap='gray')
plt.title('sobel y'),plt.xticks([]),plt.yticks([])
plt.show()
```

Output:



Roll no: 37
Name: jaimini patel

Ex no: 27

Date: 05-02-2020

Problem Statement:

Color complement using OpenCv

Program:

```
import cv2
img = cv2.imread('sharp.jpg')
comp_image = 255 - img
cv2.imshow('original image',img)
cv2.imshow("Complementary image",comp_image)
cv2.waitKey(0)
cv2.destoryAllWindows()
```

Output:

Roll no: 37
Name: jaimini patel

Ex. No: 28
Date: 05/02/2020

Problem Statement:

Color Slicing using OpenCv.

Program:

Output:

Ex. No: 29
Date: 05/02/2020

Problem Statement:

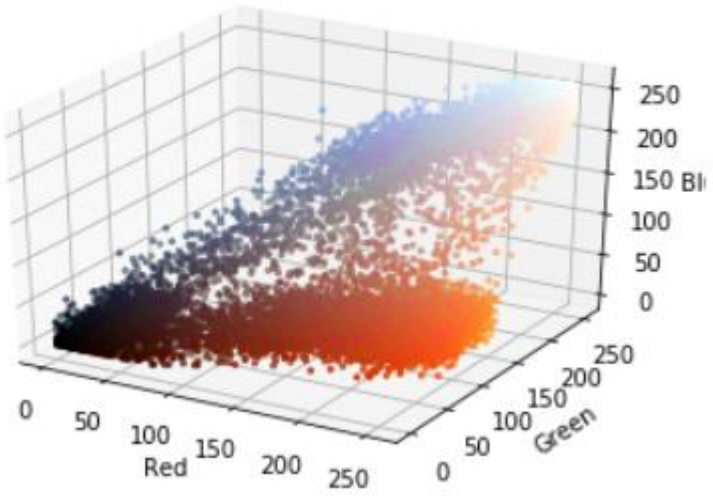
3D Scatter plot of nemo fish using OpenCv.

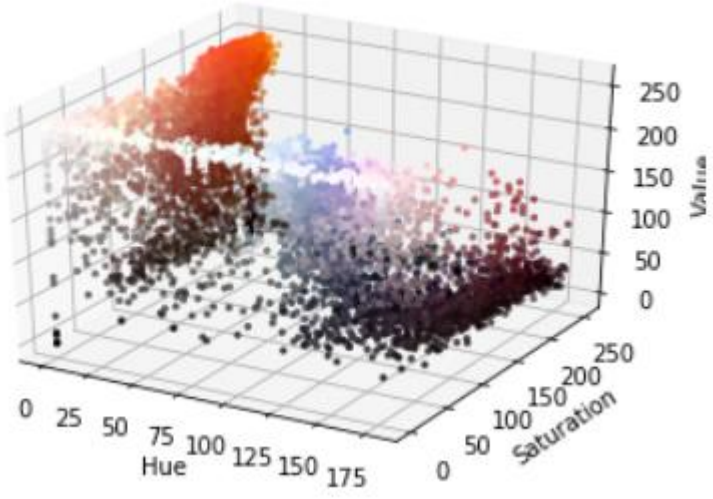
Program:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import colors
from mpl_toolkits.mplot3d import Axes3D
from matplotlib.colors import hsv_to_rgb
# To get a list of all the possible colour conversions
flags = [i for i in dir(cv2) if i.startswith("COLOR_")]
print(len(flags), "flags total:")
print(flags[40])
nemo = cv2.imread("nemo.jpg")
.imshow("original",nemo)
nemo = cv2.cvtColor(nemo, cv2.COLOR_BGR2RGB)
# Plotting the image on 3D plot
r, g, b = cv2.split(nemo)
fig = plt.figure()
axis = fig.add_subplot(1, 1, 1, projection="3d")
pixel_colors = nemo.reshape((np.shape(nemo)[0] * np.shape(nemo)[1], 3))
norm = colors.Normalize(vmin=-1.0, vmax=1.0)
norm.autoscale(pixel_colors)
pixel_colors = norm(pixel_colors).tolist()
axis.scatter(
   r.flatten(), g.flatten(), b.flatten(), facecolors=pixel_colors, marker="."
)
axis.set_xlabel("Red")
axis.set_ylabel("Green")
axis.set_zlabel("Blue")
plt.show()
```

```
hsv_nemo = cv2.cvtColor(nemo, cv2.COLOR_RGB2HSV)
h, s, v = cv2.split(hsv_nemo)
fig = plt.figure()
axis = fig.add_subplot(1, 1, 1, projection="3d")
axis.scatter(
    h.flatten(), s.flatten(), v.flatten(), facecolors=pixel_colors, marker="."
)
axis.set_xlabel("Hue")
axis.set_ylabel("Saturation")
axis.set_zlabel("Value")
plt.show()
```

Output:





Ex. No: 30
Date: 06/02/2020

Problem Statement:

Smoothing and Sharpening using OpenCv(Use Skeleton Image).

Program:

```
import numpy as np
from matplotlib import pyplot as plt
import cv2
image=cv2.imread('Capture.jpg',0)
k=image
#Original image
Laplacian=cv2.Laplacian(image,cv2.CV_64F)
plt.figure(figsize=(10,10))
plt.subplot(2,4,1)
plt.imshow(image,cmap='gray')
plt.title('Original')
plt.xticks([])
plt.yticks([])
#Laplacian filter of image
plt.subplot(2,4,2)
plt.imshow(Laplacian,cmap='gray')
plt.title('Laplacian')
plt.xticks([])
plt.yticks([])
#Sharpening Filter
k=image+Laplacian
plt.subplot(2,4,3)
plt.imshow(k,cmap='gray')
plt.title('Sharpening')
plt.xticks([])
plt.yticks([])
image=cv2.imread("Capture.jpg",0)
#Sobel Filter of original image
```

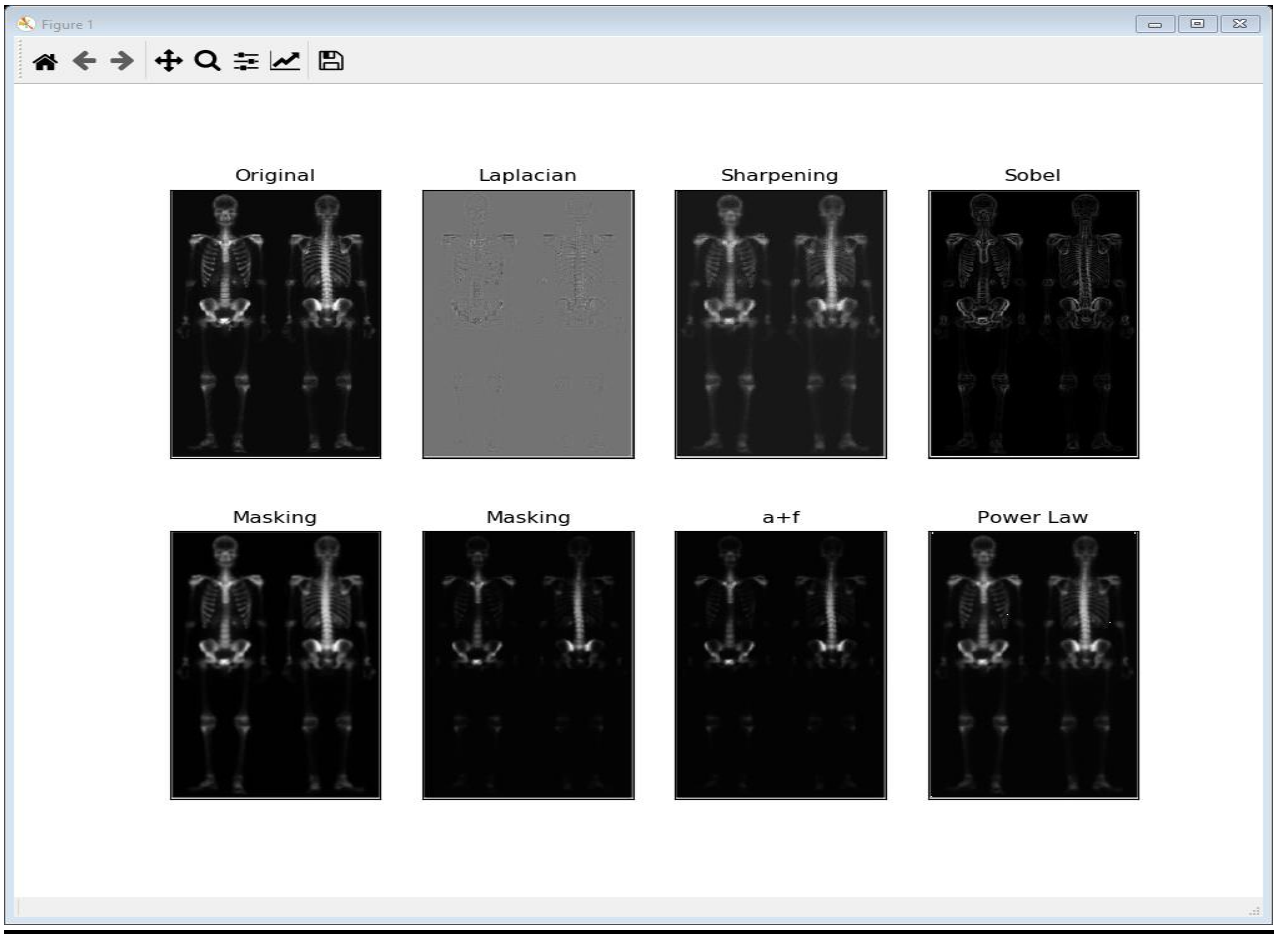
```
sobelx=cv2.Sobel(image,cv2.CV_64F,0,1,ksize=5)

sobely=cv2.Sobel(image,cv2.CV_64F,1,0,ksize=5)
sobel=np.sqrt(np.square(sobelx)+np.square(sobely))
plt.subplot(2,4,4)
plt.imshow(sobel,cmap='gray')
plt.title('Sobel')
plt.xticks([])
plt.yticks([])
#Averaging of sobel Filter
e=cv2.blur(image,(5,5))
plt.subplot(2,4,5)
plt.imshow(e,cmap='gray')
plt.title('Masking')
plt.xticks([])
plt.yticks([])
# C*E
f=k*e
plt.subplot(2,4,6)
plt.imshow(f,cmap='gray')
plt.title('Masking')
plt.xticks([])
plt.yticks([])
#g=a*f
g=image+f
plt.subplot(2,4,7)
plt.imshow(g,cmap='gray')
plt.title('a+f')
plt.xticks([])
plt.yticks([])
#Power law of g
img1=g/255
h=cv2.pow(img1,0.6)
plt.subplot(2,4,8)
plt.imshow(h,cmap='gray')
plt.title('Power Law')
plt.xticks([])
```

```
plt.yticks([])

plt.show()
cv2.imshow('Power law',h)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Output:



Roll no: 37
Name: jaimini patel

Roll no: 37
Name: jaimini patel

Ex. No: 31
Date: 06/02/2020

Problem Statement:

Spatial Filtering.

Program:

```
import cv2
import numpy as np
image=cv2.imread('elephant.jpg')
cv2.imshow('original image',image)
kernel_3x3=np.ones((3,3),np.float32)/9
blurred=cv2.filter2D(image,-1,kernel_3x3)
cv2.imshow('3*3 kernel blurring',blurred)
cv2.waitKey(0)
cv2.destoryAllWindow()
```

Output:

Roll no: 37
Name: jaimini patel

Ex. No: 32
Date: 06/02/2020

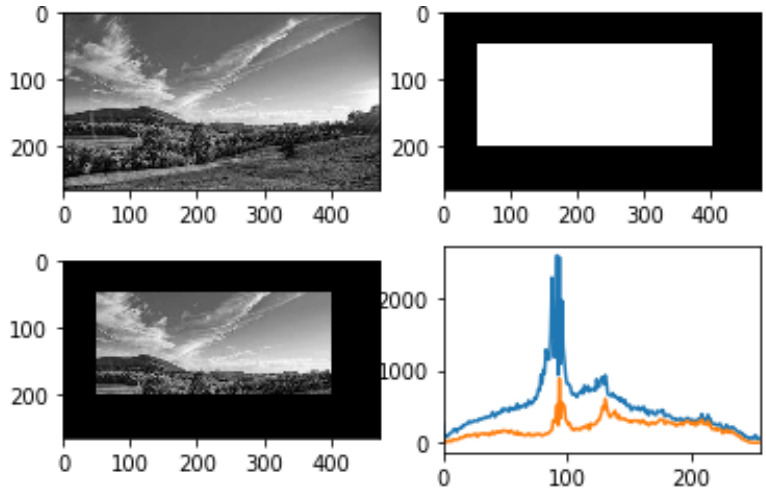
Problem Statement:

Histogram Equilization.

Program:

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
img=cv2.imread('n.jpg',0)
%matplotlib inline
mask= np.zeros(img.shape[:2],np.uint8)
mask[50:200,50:400]=255
masked_img=cv2.bitwise_and(img,img,mask=mask)
hist_full=cv2.calcHist([img],[0],None,[256],[0,256])
hist_mask=cv2.calcHist([img],[0],mask,[256],[0,256])
plt.subplot(221),plt.imshow(img,'gray')
plt.subplot(222),plt.imshow(mask,'gray')
plt.subplot(223),plt.imshow(masked_img,'gray')
plt.subplot(224),plt.plot(hist_full),plt.plot(hist_mask)
plt.xlim([0,256])
plt.show()
```

Output:



Roll no: 37
Name: jaimini patel

Ex. No: 33
Date: 06/02/2020

Problem Statement:

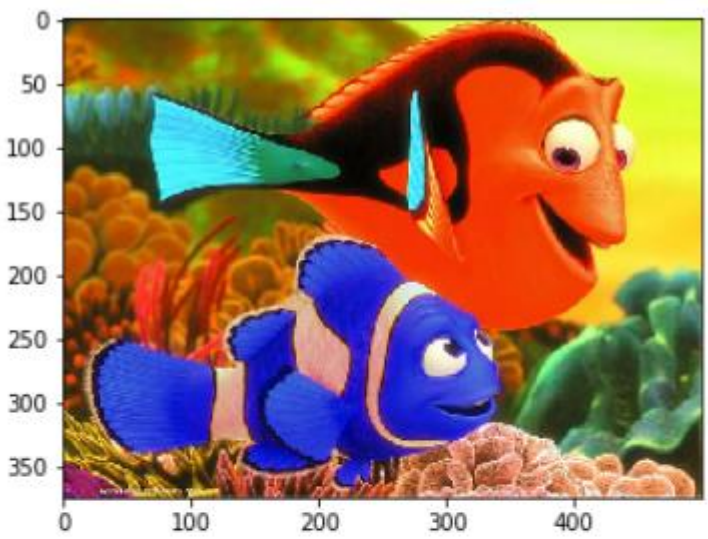
Segmentation of Color image in opencv.

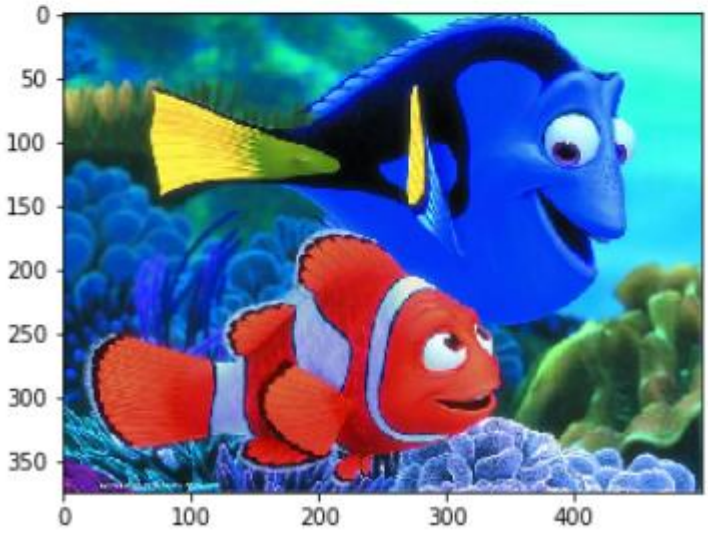
Program:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.colors import hsv_to_rgb
%matplotlib inline
nemo=cv2.imread('1.jpg')
plt.imshow(nemo)
plt.show()
nemo=cv2.cvtColor(nemo,cv2.COLOR_BGR2RGB)
plt.imshow(nemo)
plt.show()
hsv_nemo=cv2.cvtColor(nemo,cv2.COLOR_BGR2HSV)
plt.imshow(hsv_nemo)
plt.show()
light_orange=(1,190,200)
dark_orange=(18,255,255)
lo_squre=np.full((10,10,3),light_orange,dtype=np.uint8)/255.0
do_squre=np.full((10,10,3),dark_orange,dtype=np.uint8)/255.0
plt.subplot(1,2,1)
plt.imshow(hsv_to_rgb(do_squre))
plt.subplot(1,2,2)
plt.imshow(hsv_to_rgb(lo_squre))
plt.show()
mask=cv2.inRange(hsv_nemo,light_orange,dark_orange)
result=cv2.bitwise_and(nemo,nemo,mask=mask)
plt.subplot(1,2,1)
plt.imshow(mask,cmap="gray")
plt.subplot(1,2,2)
plt.imshow(result)
plt.show()
```

Roll no: 37
Name: jaimini patel

Output:





Roll no: 37
Name: jaimini patel