Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

Lab Report 1

COMP 342

(For partial fulfillment of 3rd Year/ 2nd Semester in Computer Engineering)

Submitted to:

Mr. Dhiraj Shrestha

Department of Computer Science and Engineering

Submitted by:

Neha Malla

C.E.

Roll No. 27

1. Mention the name of Programming language and Graphics Library you are using this semester for performing your Computer Graphics Lab and Project

- ➔ **Programming language:** Python
- ➔ **Graphics library:** Pygame and PyOpenGL

2. Write the code snippets for setting graphics environment in your chosen graphics library and display the resolution of your display system through functions/classes provided by your graphics library

- ➔ **To install:** Pip install pygame
- ➔ **To import:** import pygame, import pygame.gfxdraw
- ➔ **Code snippet to display the resolution:**

```
import pygame

pygame.init()



screen = pygame.display.set_mode([0,0])

w, h = pygame.display.get_surface().get_size()

print("Resolution: ", w, "X", h)
```

3. Get Familiar with the coordinate system and Draw a flag of Nepal using the chosen Graphics geometrical functions/ classes provided by the your chosen graphics library and also color the flag accordingly

- ➔ **Source Code:**

```
import pygame

import pygame.gfxdraw

from math import sin, cos, pi
```

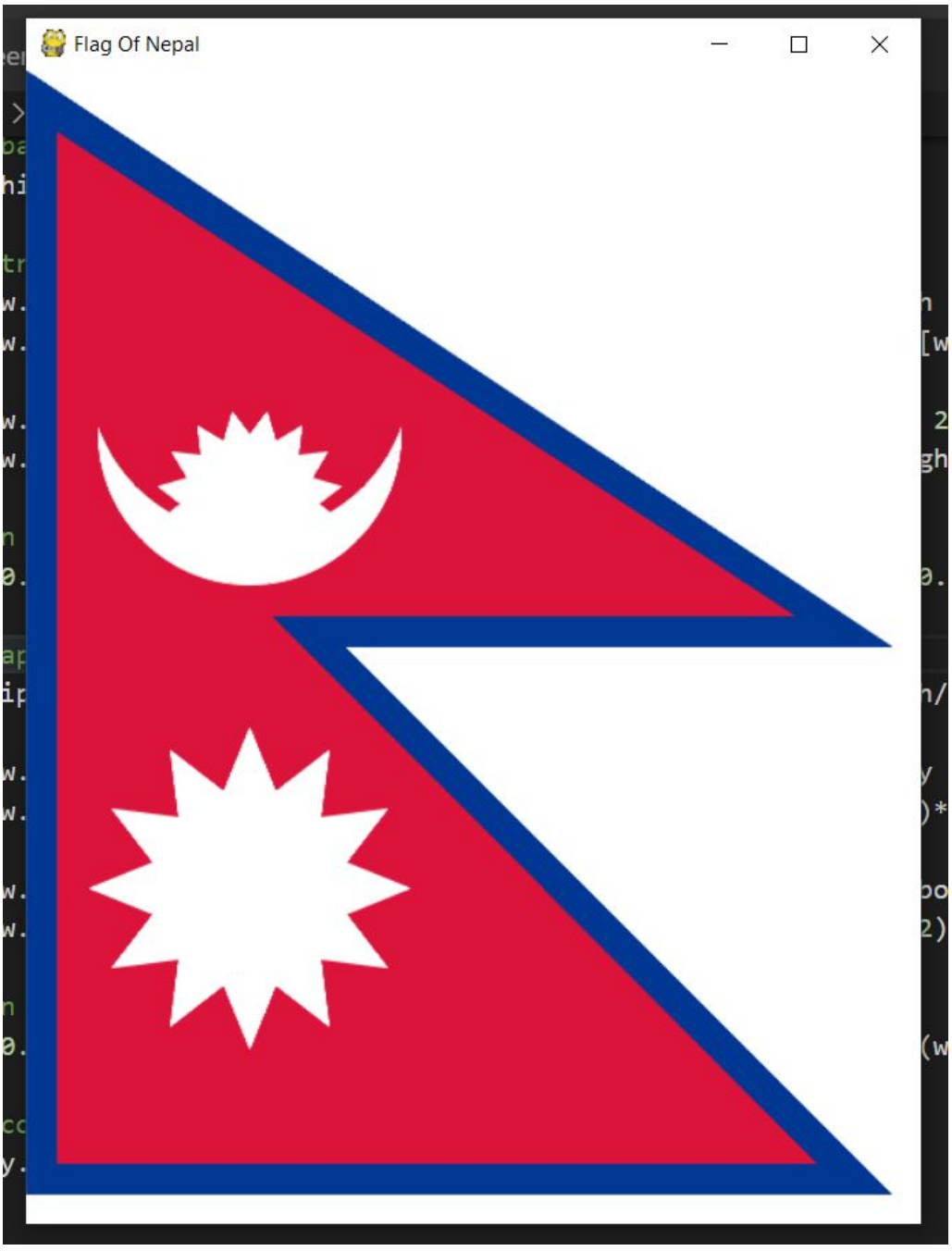
```
# Initializing the game engine

pygame.init()



# Colors in RGB format

white = (255, 255, 255)

blue = (0, 56, 147)

crimson = (220, 20, 60)



# height and width of the screen, border of flag

width = 450

height = int((4/3)*width)

border = int(width/24) #20

screen = pygame.display.set_mode([width + 4*border, height + 4*border],
pygame.RESIZABLE)

pygame.display.set_caption("Flag Of Nepal")



# The inner coordinates of the flag are:

# C(0,0), A(0, height), B(width, height), E(width - sin(45)*width, height
- sin(45)*width), G(width, height - sin(45)*width)

Ex = int(width - sin(pi/4)*width) + border

EGy = int(height - sin(pi/4)*width) + 2*border



# Function for sun shape

def SunShape(xc, yc, rc, rt, point):
```

```
    if point == 12:
        sunArr=[[xc + rt, yc]]
        angle = 0
        inc = pi/12
    elif point == 8:
        sunArr=[[xc, yc + rc], [xc + int(rt*cos(pi-pi/16)), yc +
int(rt*sin(pi-pi/16))]]
        angle = pi-pi/16
        inc = pi/16

    while angle < 2*pi - pi/6:
        angle += inc
        sunArr +=  [[xc + int(rc*cos(angle)), yc + int(rc*sin(angle))]]
        angle += inc
        sunArr +=  [[xc + int(rt*cos(angle)), yc + int(rt*sin(angle))]]
        if point ==8 and angle >= 2*pi - pi/6:
            angle += inc
            sunArr +=  [[xc + int(rc*cos(angle)), yc +
int(rc*sin(angle))]]
            angle += inc
            sunArr +=  [[xc + int(rt*cos(angle)), yc +
int(rt*sin(angle))]]

    pygame.gfxdraw.aapolygon(screen, sunArr, white)
    pygame.gfxdraw.filled_polygon(screen, sunArr, white)
```

```
done = False


while not done:
    # If user clicks close
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            done = True


    # The screen background as white
    screen.fill(white)


    # The double triangular flag shape
    pygame.gfxdraw.aapolygon(screen, [[0, 0], [0, height + 3*border], [width + 3*border, height + 3*border], [Ex + 2*border, EGy + border], [width + 3*border, EGy + border]], blue)
    pygame.gfxdraw.filled_polygon(screen, [[0, 0], [0, height + 3*border], [width + 3*border, height + 3*border], [Ex + 2*border, EGy + border], [width + 3*border, EGy + border]], blue)


    pygame.gfxdraw.aapolygon(screen, [[border, 2*border], [border, height + 2*border], [width + int(0.60*border), height + 2*border], [Ex - int(0.40*border), EGy], [width - int(0.20*border), EGy]], crimson)
    pygame.gfxdraw.filled_polygon(screen, [[border, 2*border], [border, height + 2*border], [width + int(0.60*border), height + 2*border], [Ex - int(0.40*border), EGy], [width - int(0.20*border), EGy]], crimson)
```

```
    # The full sun shape

    SunShape(int(0.25*width) + border, EGy + int(0.5*width*sin(pi/4)),
int(0.21*(EGy - 2*border)),  int((EGy - 2*border)/3), 12)



    # The moon shape

    screen.set_clip(2*border, int(EGy/2 + border), int(width/2.4),
int(width/5.1 + 2*border))# l = 2.55



    pygame.gfxdraw.aacircle(screen, int(0.25*width) + border,
int((5/8)*(EGy - 2*border)) + 2*border, int(width/5.1), white) # radius =
5.25

    pygame.gfxdraw.filled_circle(screen, int(0.25*width) + border,
int((5/8)*(EGy - 2*border)) + 2*border, int(width/5.1), white)



    pygame.gfxdraw.aacircle(screen, int(0.25*width) + border, int(EGy/2) +
border, int((EGy - 2*border)/3), crimson)

    pygame.gfxdraw.filled_circle(screen, int(0.25*width) + border,
int(EGy/2) + border, int((EGy - 2*border)/3), crimson)



    # The half sun shape

    SunShape(int(0.25*width) + border, int((3/8)*height + 1.5*border),
int(width/11),  int(width/8.16), 8) # rc = 10.2



    # Update the contents of the entire display

    pygame.display.flip()



pygame.quit()
```

➔ **Output:**



➔ **Conclusion:**

The dimensions for the flag were calculated as stated in the step by step procedure in the constitution of Nepal, with slight alterations when necessary. A Python library called Pygame was used to draw and display the flag.