

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



ЛАБОРАТОРНА РОБОТА №3
З ДИСЦИПЛІНИ:
«Комп'ютерна графіка»

ВАРІАНТ №8

Виконав:

ст. гр. ІР-31

Касараба В.В.

Прийняла:

асистент. каф. КСА

Шпак О.І.

Львів 2021

ЗАВДАННЯ

Написати програму мовою C++, яка буде за заданим алгоритмом на екрані растеризоване зображення згідно отриманого варіанту. Варіанти завдань беруть з таблиці 1 за вказівкою викладача.

(Можливе використання довільної мови програмування за вибором студента)

Основні вимоги:

1) Всі завдання (окрім завдань із зафарбовуванням областей) необхідно виконувати, використовуючи псевдопіксель, у вигляді зафарбованого суцільним кольором квадрата розміром $k \times k$ пікселів.

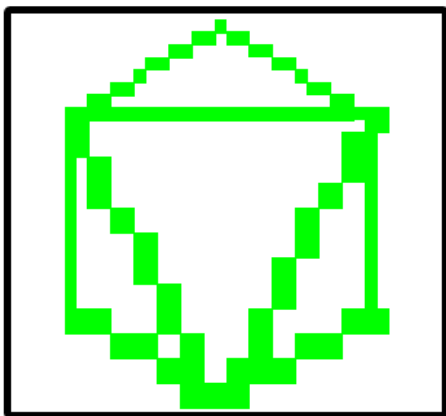
2) Всі алгоритми растеризації реалізувати у вигляді функції із необхідними параметрами. Наприклад, функція побудови лінії за методом Брезенхейма

`BresenhamLine(x1, y1, x2, y2, color)`

де x_1, y_1 - точки початку відрізка, x_2, y_2 - точки кінця відрізка, $color$ - колір відрізка.

3) Необхідно забезпечити, щоб розміри зображення змінювалися при зміні розмірів області виведення (вікна екрану).

8



Методом цифрового диференціального аналізатора побудувати відрізки зеленого кольору, що утворюють найбільше зображення октаедра, яке поміститься в області екрану. Зовнішній контур зображення октаедра є **правильним шестикутником**. Сторони октаедра розміщені найближче до спостерігача (жирні лінії) побудувати псевдопікселями розміром 20×20 пікселів. Віддалені сторони необхідно побудувати псевдопікселями розміром 10×10 пікселів.

Виконання роботи

Python Code

```
import pygame
import sys

WIN_WIDTH = 700
WIN_HEIGHT = 570

WHITE = (255, 255, 255)
BLACK = (0, 0, 0)
GREEN = (0, 235, 20)

sc = pygame.display.set_mode((WIN_WIDTH, WIN_HEIGHT))

def draw(color):
    sc.fill(WHITE)

    x_top = WIN_WIDTH // 2
    pygame.draw.rect(sc, color, pygame.Rect(x_top-10, 0, 20, 10))
    for i in range(11):
        pygame.draw.rect(sc, color, pygame.Rect(x_top+20*i, 10+10*i, 20, 10))
        pygame.draw.rect(sc, color, pygame.Rect(x_top-20-
20*i, 10+10*i, 20, 10))

        pygame.draw.rect(sc, color, pygame.Rect(540-20*i, 130+40*i, 20, 40))
        pygame.draw.rect(sc, color, pygame.Rect(140+20*i, 130+40*i, 20, 40))

        if i < 6:
            pygame.draw.rect(sc, color, pygame.Rect(x_top-20+40*i, 550-
20*i, 40, 20))
            pygame.draw.rect(sc, color, pygame.Rect(x_top-20-40*i, 550-
20*i, 40, 20))

        pygame.draw.rect(sc, WHITE, pygame.Rect(x_top-10, 10, 20, 10))
        pygame.draw.rect(sc, color, pygame.Rect(130, 110, 420, 10))

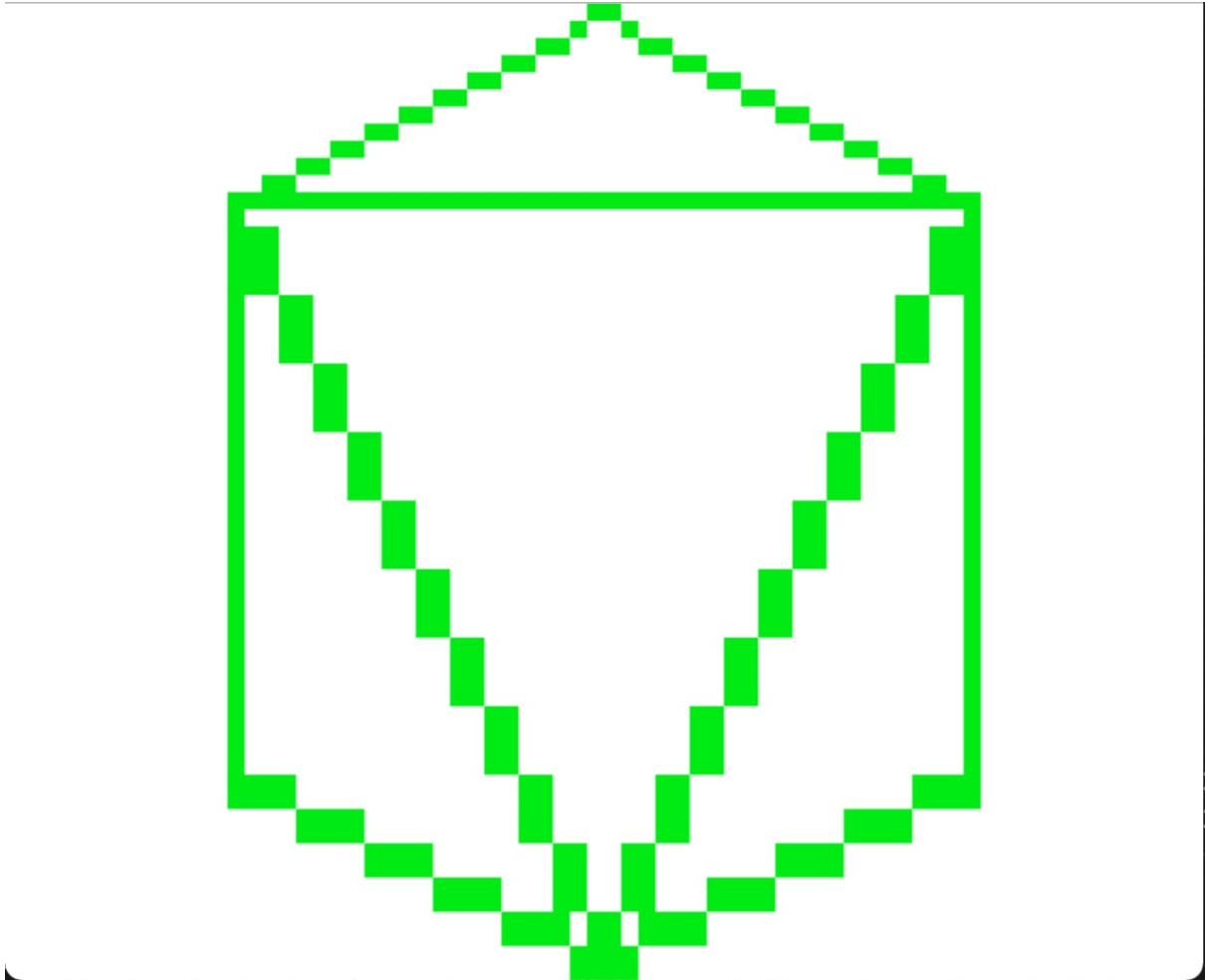
        pygame.draw.rect(sc, color, pygame.Rect(130, 110, 10, 350))
        pygame.draw.rect(sc, color, pygame.Rect(560, 110, 10, 350))

    pygame.display.update()

if __name__ == '__main__':
    while 1:
```

```
draw(GREEN)
for i in pygame.event.get():
    if i.type == pygame.QUIT:
        sys.exit()
    if i.type == pygame.KEYDOWN:
        draw(GREEN)
```

Результат виконання



Висновок:

На цій лабораторній роботі я навчився писати програму мовою Python, яка будує за заданим алгоритмом на екрані растроване зображення згідно отриманого варіанту.

Код програми та результат роботи я вніс у звіт.