25.10.2023

Grafika komputerowa i komunikacja człowiek komputer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Prowadzący | Autor | Kurs | Godzina Zajęć | Tytuł Laboratorium |
| Tomasz Zamojski | Bartłomiej Muszyński 269196 | Grafika Komputerowa | Środa 14:15 | Podstawy OpenGL |

Spis Treści

[1. Temat Laboratorium 3](#_Toc150327867)

[2. Zadania do wykonania 3](#_Toc150327868)

[3.0 3](#_Toc150327869)

[3.5 3](#_Toc150327870)

[4.0 3](#_Toc150327871)

[4.5 3](#_Toc150327872)

[5.0 3](#_Toc150327873)

[3. Wykonanie zadań 4](#_Toc150327874)

[3.0 4](#_Toc150327875)

[3.5 6](#_Toc150327876)

[4.0 8](#_Toc150327877)

# Temat Laboratorium

Celem tego laboratorium było zapoznanie się z biblioteką OpenGL oraz wykonanie podstawowych zadań(1-5) zgodnie z zaleceniami.

# Zadania do wykonania

Do wykonania zostało zlecone pięć zadań z wykładu:

## 3.0

Przygotowanie środowiska pracy.

## 3.5

Napisanie funkcji rysującej prostokąt w podanym miejscu.

## 4.0

Do wcześniej napisanej funkcji należy wprowadzić losowość kolorów i deformacje prostokąta.

## 4.5

Należy narysować fraktal – prostokątny dywan Sierpińskiego.

## 5.0

Należy narysować kolejny fraktal.

Szczegóły dotyczące zadań można znaleźć na stronie https://datko.pl/GK/ w opdowiednim wykładzie.

# Wykonanie zadań

W tym punkcie znajduje się opis wprowadzonych przezemnie zmian w kodzie źródłowym które miały na celu wykonanie poszczególnych zadań.

## 3.0

W ramach tego zadania odwiedziłem stronę <https://www.python.org/downloads/windows/>, pobrałem odpowiedni plik i zainstalowałem go zgodnie z zaleceniami, następnie zainstalowałem bibliotekę openGL oraz glfw przy użyciu komendy pip install.

W następnym kroku pobrałem kod źródłowy programu i zweryfikowałem jego działanie.

Na końcu mogłem przejźć do faktycznego zadania, sprawienia aby każdy wierzchołek miał swój własny kolor.

Aby to osiągnąć usunąłem jeden z dwóch bazowych trójkątów oraz użyłem glColor3f dla każdego z wierzchołków.

W celu uzyskania większej kontroli nad trójkątem dodałem również zmienne center\_x = 0 center\_y = 0 sterujące środkiem trójkąta oraz side\_length = 50.0 żeby móc stworzyć trójkąt równoboczny (code1).

Wynik działania programu można znaleźć w obrazku (image1).

A screen shot of a computer program

Description automatically generated

(code1)

A screenshot of a computer

Description automatically generated

(image1)

## 3.5

Przy tym zadaniu musiałem przywrócić wcześniej usunięty trójkąt do funkcji render().

Po jego odyzskaniu poprzez użycie zmiennych center\_x oraz center\_y uzyskałem kwadrat więc następnym krokiem było wprowadzenie dwóch dłuższych boków, w celu ich uzyskania wprowadziłem zmienne width = 100.0 oraz height = 50.0 zastępujące side\_length z poprzedniego zadania (code2).

Rezultat działania programu można znaleźć w poniższym obrazku (image2).

A screen shot of a computer program

Description automatically generated

(code2)

A screenshot of a computer screen

Description automatically generated

(image2)

## 4.0

Zgodnie z poleceniem zaimportowałem bibliotekę random i pierwsze co zrobiłem to wylosowanie dwunastu kolorów oraz jednej deforamacji w funkcji startup() oraz ustawienie ich jako zmiennych globalnych(code3).

A screen shot of a computer

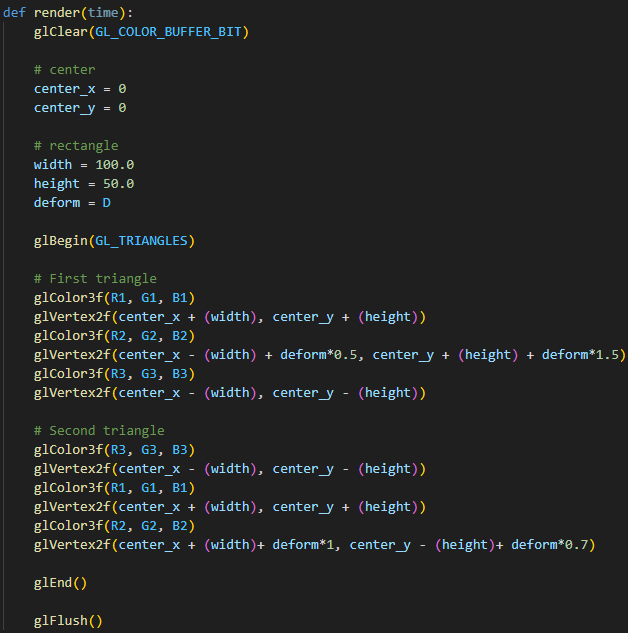
Description automatically generated

(code3)

Następnie przeszedłem do modyfikacji prostokąta w funkcji render() co okazało się dość intuicyjne.

Zmieniłem stałe wartości kolorów na zmienne zwracając uwagę na bok „łączący” dwa trójkąty oraz dodałem deformacje do dwóch różnych wierzchołków prostokąta(code4).

Wynik działania programu można znaleźć na obrazku (image3)



(code4)

A screenshot of a computer

Description automatically generated

(image3)