**SPRAWOZDANIE**

Zajęcia: Grafika komputerowa

Prowadzący: prof. dr hab. Vasyl Martsenyuk

**Laboratorium: Grafika Komputerowa**

06.04.2020

**Temat: „**Światło i materiały w OpenGL”

**Wariant:** -

Michał Krzyżowski

Informatyka I stopień,

stacjonarne,

4 semestr,

Gr.1b

**1. Polecenie:**

W obecnych laboratoriach należało stworzyć piramidę (w poleceniu nie jest sprecyzowane jakim ostrosłupem ma ona być) z użyciem różnych materiałów i umieszczeniem jej na „podstawie”. Po narysowaniu widać że włączone jest jedynie oświetlenie podstawowe. W ramach laboratorium należało poprawić te oświetlenie. Do wyboru był języka Java i C.

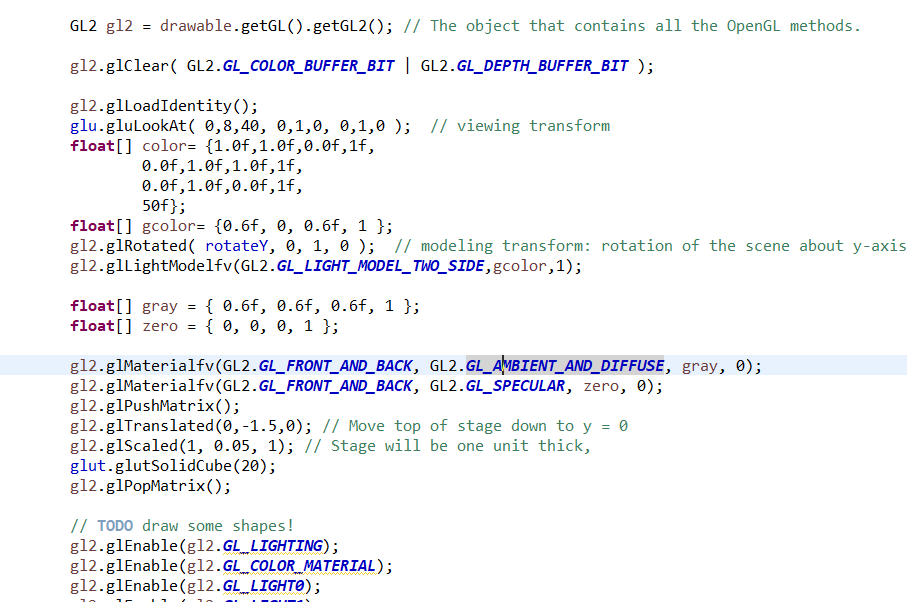
**2. Wprowadzane dane:**

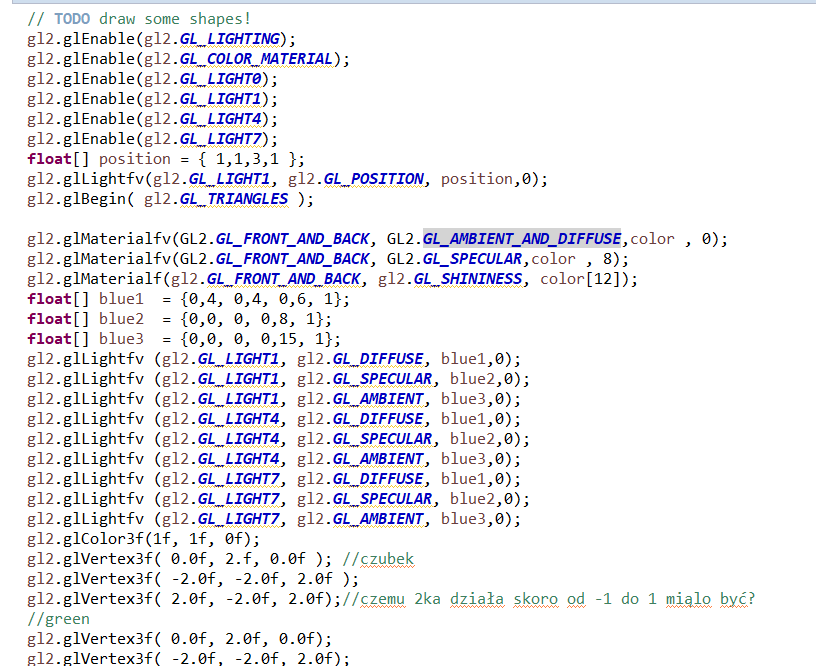
Dane w zadaniu ustawione są na stałe (znajdują się bezpośrednio w kodzie) są to choćby materiały jakie są używane do budowy piramidy, ustawienia kolorów i światła. Po uruchomieniu samego programu już nie wprowadzamy żadnych danych, poza tym iż możliwe jest poinformowanie aplikacji że chcemy przesunąć obraz, na co sama aplikacja zareaguje.

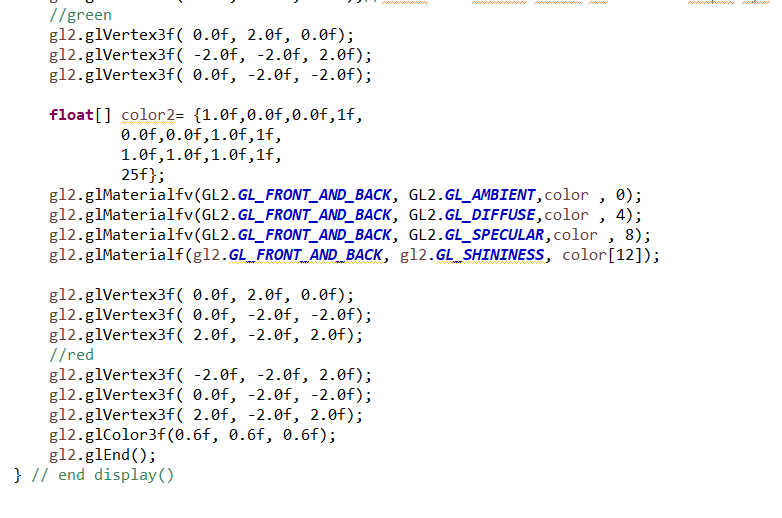
**3. Wykorzystane komendy:**

Komendy wykorzystywane podczas pracy nad projektem to instrukcje: glVertex3f która służy do ustawienia punktu w przestrzeni 3d. Dzięki tej komendzie możliwe było rysowanie brył 3d gdyż składają się one z wierzchołków które połączone są ze sobą tworząc całość. Kolejna komenda to glMaterialfv(), przyjmuje ona 3 lub 4 parametry. W moim kodzie przeładowanie przyjmujące 4-ry parametry jest wykorzystywane. Pierwszy parametr informuje o stronie bryły na którą ma to działać – frontowa/ tylnia/ tylnia i fronotwa, drugi parametr mówi o właściwościach materiału dostępne opcje to GL\_Ambient/ gl\_diffuse, gl\_specular, gl\_emission albo też połączenie dwóch z nich na raz gl\_ambient\_and\_diffuse, trzeci parametr to tablica dla kolorowych komponentów, która jeżeli dobrze zrozumiałem jest także używana przy obliczaniu oświetlenia. W moim kodzie rozbiłem to na 4 osobne komendy, które kolejno ustawiają poprzez te tablice kolejno kolor otoczenia, kolor rozproszony, kolor otoczenia i ostatnia pojedyncza wartość odpowiadająca za połysk. Kolejna komenda to glenable która włącza dane światło, podając parametr np. gl\_light0/ gl\_light1. Kolejna komenda tyczy się wcześniej aktywowanego światła, zmiany jego właściwości jest to glLightfv(). Funkcja ta przyjmuje kolejno jako parametr pierwszy światło które ma być skonfigurowane, drugi informuje która z właściwości światła ma ulec konfiguracji (tutaj możemy ustawić pozycje danego światła), 3-ci parametr tyczy się koloru. Komenda gllightmodelfv() odpowiada za ustawienia globalne światła.

Poniższy kod prezentuje rysowanie piramidy, próby konfigurowania ustawień światła:





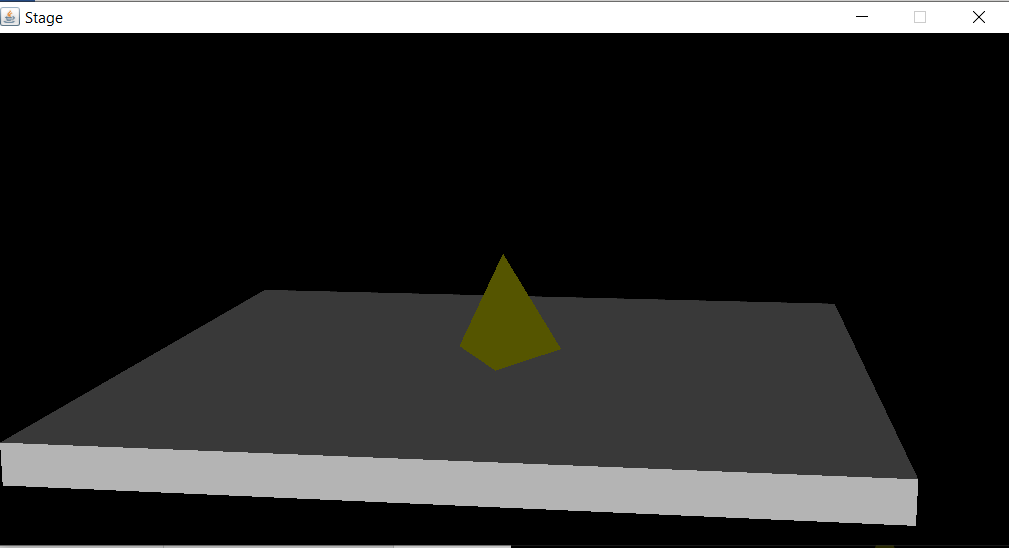
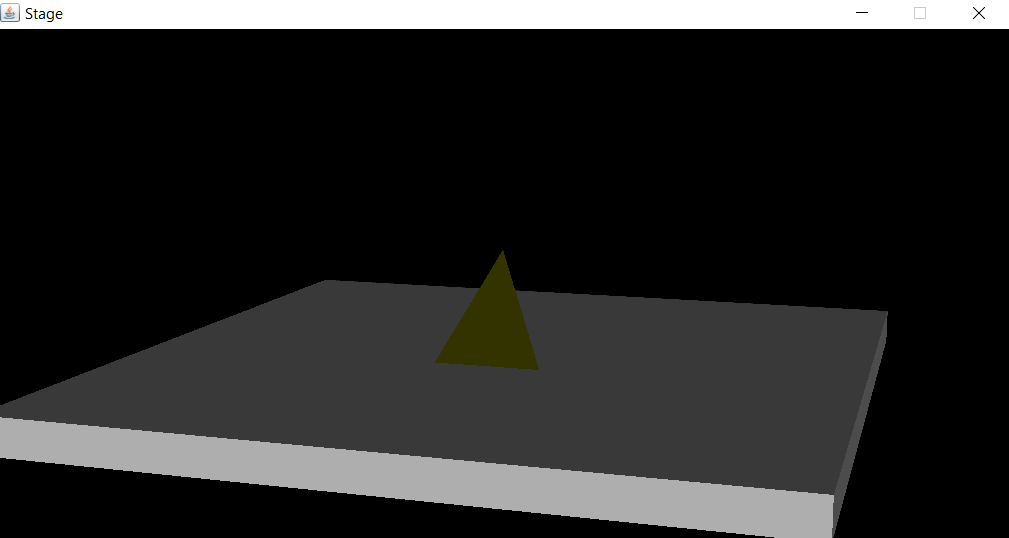
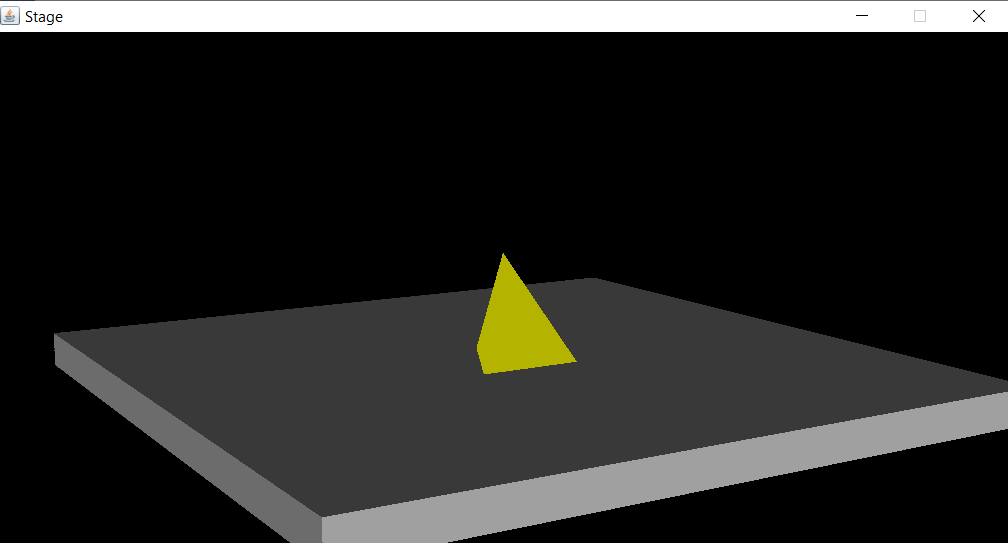
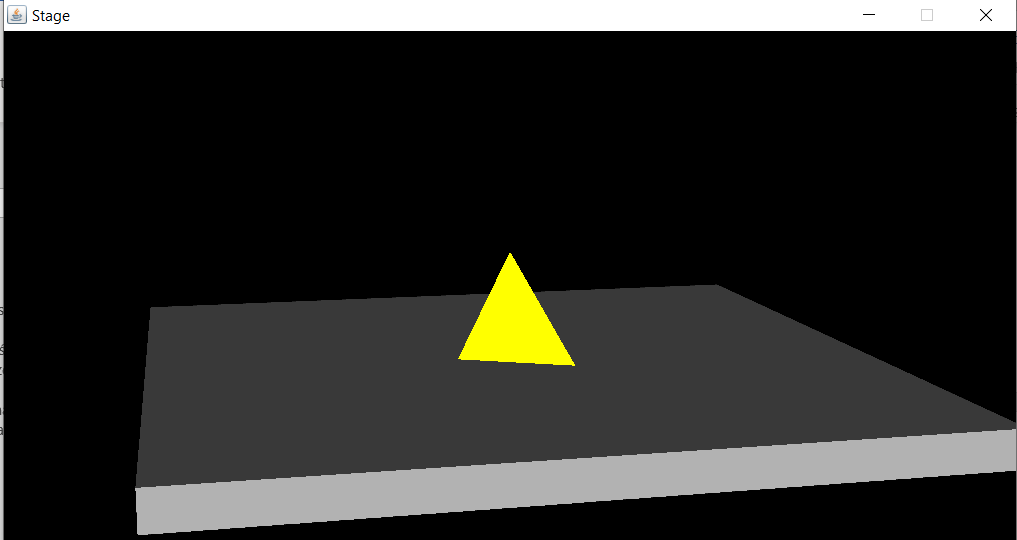


Link do zdalnego repozytorium zawierającego projekt:

<https://github.com/Jarverr/GrafikaKomputerowaLab6>

**4. Wynik działania:**

Zrzuty ekranu prezentujące wyniki zadania (piramida z różnych perspektyw):



**5. Wnioski:**

Dzięki możliwością jakie daje biblioteka OpenGL, jestem w stanie w stosunkowo prostu sposób stworzyć trójwymiarową bryłę a następnie sformatować światło, które na nią pada. Niestety z własnych doświadczeń, mam wrażenie, że operowanie tymi światłami jest mało intuicyjne, choć może to wynikać z braku doświadczenia. Na pewno możliwości modyfikacji światła jest wiele, tako samo ustawień właściwości obiektu/ bryły jak powinna się zachowywać gdy spotyka się ze światłem. Czyni to całkiem solidne narzędzie do operowania zachowaniem światła w grafice.