

# Sprawozdanie

## Projekt 2: Eliminacja elementów zasłoniętych

Paweł Drapiewski 271050

Adam Gościcki 257836

### Cel

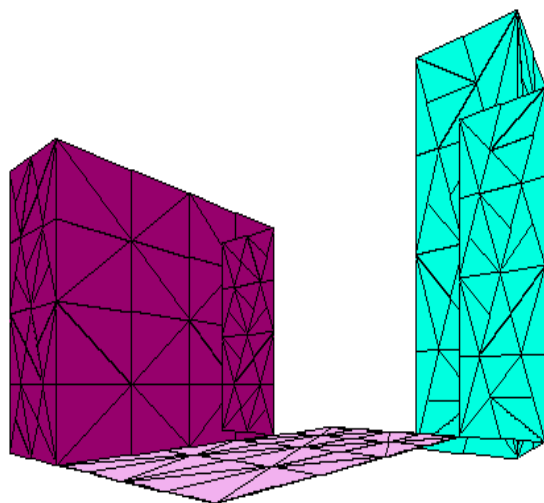
Celem projektu było zaimplementowanie do istniejącego już kodu algorytmu, który pozwoli na eliminację zasłoniętych ścian, aby obiekty mogły być postrzegane jako bloki, a nie zbiór losowo rysowanych linii.

### Implementacja

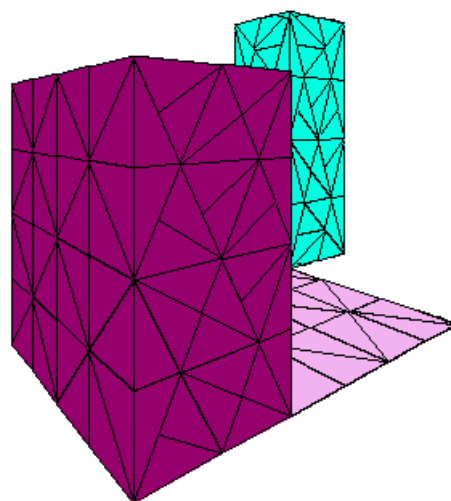
Projekt wykonaliśmy w języku Python przy pomocy bibliotek:

- numpy – operacje na macierzach
- pygame – rysowanie, kolorowanie oraz wypełnianie obiektów
- math – stałe matematyczne i operacje na funkcjach trygonometrycznych

Do eliminacji elementów zasłoniętych postanowiliśmy użyć algorytmu malarskiego, który polega na posortowaniu rysowanych przez nas obiektów względem odległości od kamery tak, że obiekty znajdujące się dalej będą rysowane jako pierwsze i w rezultacie będą przykryte przez obiekty występujące później. Ponieważ sposób ten jest bardzo niedokładny i zbyt prosty w implementacji dodaliśmy podział każdego poligonu (ściany obiektu) na znacznie mniejsze trójkąty, i sortowanie odbywa się dopiero na nich, względem ich środka ciężkości.



rys. 1



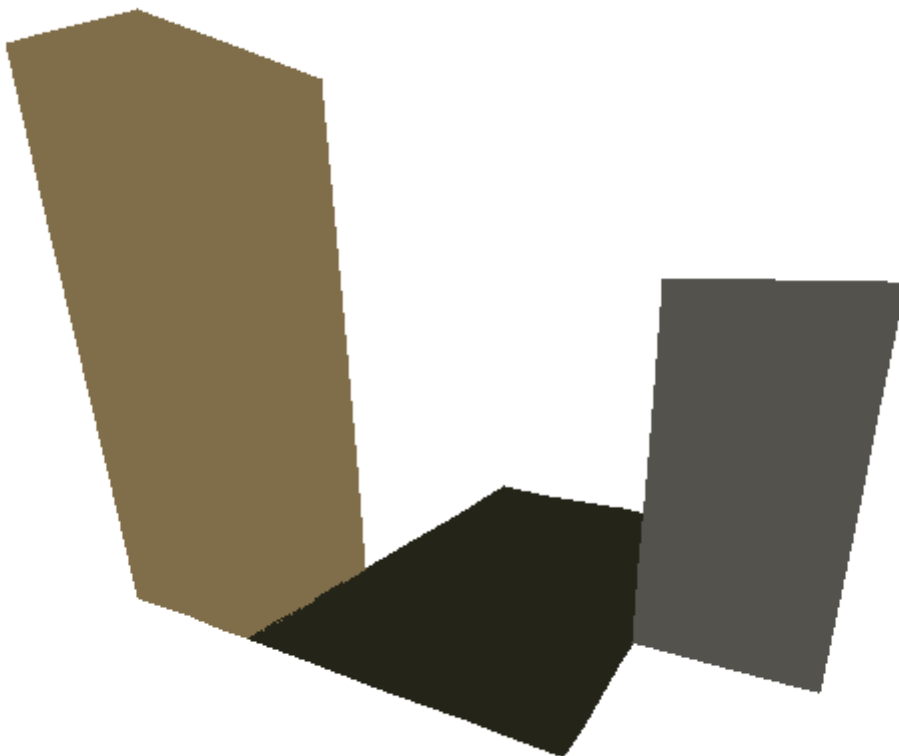
rys. 2

(rys. 1) podział na trójkąty bez zaimplementowanego algorytmu sortującego  
(rys. 2) pełna implementacja algorytmu z podziałem na mniejsze figury.

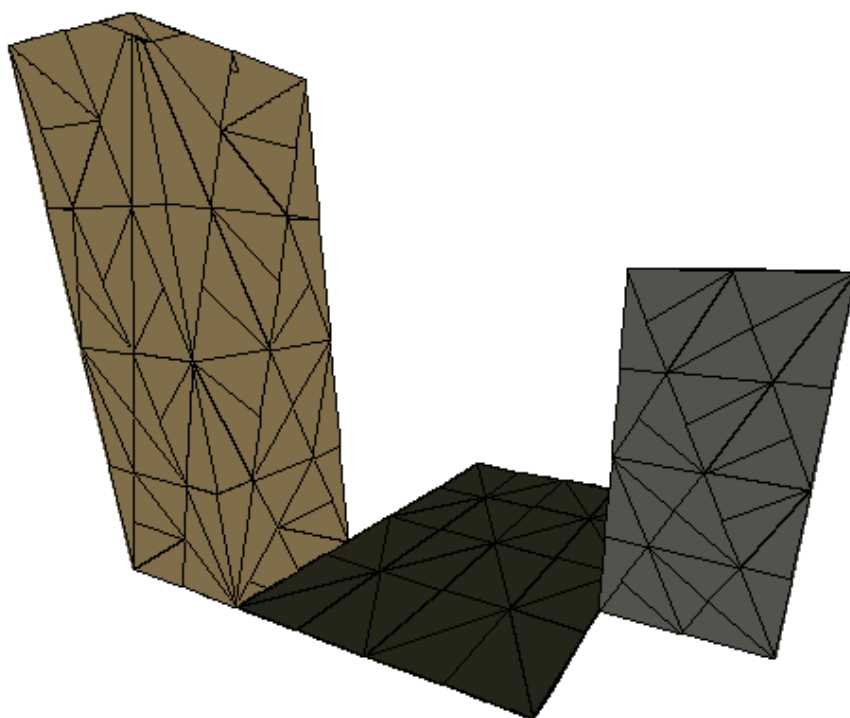
Kamera posiada zaimplementowaną klawiszologię:

- w – krok w przód
- s – krok w tył
- a – krok w lewo
- d – krok w prawo
- z – przybliżenie zoomem
- x – oddalenie zoomem
- p – krok w górę
- l – krok w dół
- strzałka w lewo – rotacja w lewo wokół osi y
- strzałka w prawo – rotacja w prawo wokół osi y
- strzałka w dół – rotacja w górę wokół osi x
- strzałka w górę – rotacja w dół wokół osi x
- q – rotacja w lewo wokół osi z
- e – rotacja w prawo wokół osi z
- o – włączenie/wyłączenie widoczności linii tworzących poligony

## Prezentacja



rys. 3 – Pełna implementacja



rys. 4 – Włączenie widoczności linii