

Zadania do wykonania (1)

Na ocenę **3.0** należy wprowadzić obracanie obiektu wokół osi X.

Wskazówki:

- przestudiować w jaki sposób zrealizowano obrót wokół osi Y o kąt `theta`,
- dodać zmienną pomocniczą `phi` i obsłużyć ruch myszą w pionie,
 - ▶ nadal bazować tylko na wciśniętym lewym przycisku myszy,
- intuicyjnie: obrót wokół osi X to obrót “góra-dół”,
- wymagane będzie zmodyfikowanie funkcji:
 - ▶ `mouse_motion_callback()`,
 - ▶ `render()`.

Zadania do wykonania (2)

(po zrealizowaniu zadania poprzedniego)

Na ocenę **3.5** należy wprowadzić obsługę drugiego przycisku myszy.

Wskazówki:

- obsługa nowego przycisku – w funkcji `mouse_button_callback()`,
- postać funkcji `mouse_motion_callback()` nie ulegnie zmianie,
- wprowadzić zmienną pomocniczą `scale`,
- w funkcji `render()`:
 - ▶ wykonać zmianę wartości zmiennej `scale`,
 - ▶ przy ruchu z wciśniętym prawym przyciskiem myszy,
- użyć wartości `scale` do przeskalowania obiektu – `glScalef()`.

Zadania do wykonania (3)

(po zrealizowaniu zadania poprzedniego)

Na ocenę **4.0** należy zrealizować poruszanie kamerą wokół modelu.

Wskazówki:

- nie powinno być konieczności modyfikacji funkcji obsługujących zdarzenia,
- zakomentować funkcje `glRotatef()` i `glScalef()` w funkcji `render()`,
- wykorzystać wartości zmiennych `theta` i `phi` do obliczenia x_{eye} , y_{eye} i z_{eye} ,
- zdarzenia prawego przycisku myszy użyć do zmiany wartości parametru `R`,
- użyć wartości x_{eye} , y_{eye} i z_{eye} jako argumentów funkcji `gluLookAt()`,
- wartości `theta` i `phi` przeskalować w funkcjach `sin()` i `cos()` przez $\frac{\pi}{180}$.

Zadania do wykonania (4)

(po zrealizowaniu zadania poprzedniego)

Na ocenę **4.5** należy usprawnić poruszanie kamerą wokół modelu.

Wskazówki:

- wprowadzić ograniczenia w zakresie przybliżania/oddalania kamery,
- zapewnić poprawność przejść kamery wokół modelu,
 - ▶ w szczególności zwrócić uwagę co dzieje się “nad” i “pod” obiektem,
 - ▶ pomoce może być użycie modułu (funkcja `fmod()` lub operator `%`) w celu ograniczenia wartości kątów do przedziału $[0; 2\pi]$,
- wprowadzić możliwość przełączania między trybem obracania obiektu i trybem poruszania kamerą – na przykład za pomocą klawiatury.

Zadania do wykonania (5)

(po zrealizowaniu zadania poprzedniego)

Na ocenę **5.0** należy zrealizować zadanie dodatkowe.

Wskazówki:

- wybrać jeden z przykładów zaproponowanych jako "zadania domowe",
 - ▶ dokument znajduje się na stronie prowadzącego,
 - ▶ uściślenie odnośnie Zadania 4.5 (Trójkąt Sierpińskiego):
 - ruch ma działać analogicznie, jak w opisie poniżej,
- alternatywnie można wykonać swobodny pierwszoosobowy ruch kamery,
 - ▶ cel: ruch podobny do znanego z gier pierwszoosobowych,
 - ▶ wprowadzić zmienną pomocniczą dla kierunku patrzenia,
 - ▶ klawisze [w][s][a][d] posłużą do zmiany położenia obserwatora,
 - ▶ ruch myszką powinien wpływać na kierunek patrzenia,
 - ▶ punkt zainteresowania / kierunek patrzenia można wyznaczyć analogicznie do przypadku z kamerą poruszaną dookoła obiektu.

Zadania domowe

Ćwiczenie 4

Rzutowanie perspektywiczne, interakcja

Zadanie 4.1

Model terenu na bazie fraktala plazmowego

-

Zadanie do wykonania

Napisać program pozwalający na wizualizację „lotu samolotem” nad nieskończenie rozległym terenem zbudowanym przy pomocy algorytmu fraktala plazmowego. Interakcja powinna zapewniać możliwość zmiany kierunku przemieszczania się i wysokości położenia obserwatora znajdującego się w kabinie hipotetycznego samolotu.

Zadanie 4.2

Symulacja ruchu planet

Zadanie do wykonania

Napisać program umożliwiający obserwację ruchu planet z punktu widzenia obserwatora znajdującego się:

- w dowolnym punkcie przestrzeni,
- na jednej z planet.

Zadanie 4.3

Torus

Zadanie do wykonania

Napisać program pozwalający na interakcyjne kształtowanie i obserwację łańcucha wykonanego z torusów.

Zadanie 4.4

Zadanie do wykonania

Napisać program umożliwiający interakcyjną modyfikację siatki punktów kontrolnych definiujących powierzchnię Beziera i pozwalający na obserwację powierzchni z różnych punktów widzenia.

Zadanie 4.5

Trójkąt Sierpińskiego – wersja 3-D

Zadanie do wykonania

Napisać program pozwalający na przemieszczanie obserwatora wewnątrz bryły powstałej w wyniku realizacji algorytmu „trójkąta Sierpińskiego 3-D”.

Zadanie 4.6

Powierzchnia oparta na funkcji Weierstrassa

Zadanie do wykonania

Napisać program umożliwiający wizualizację powierzchni obiektu opartego na funkcji Weierstrassa z możliwością dodawania w obrazie nowych szczegółów w przypadku, gdy obserwator będzie się zbliżał do powierzchni.