

Przygotowany przeze mnie projekt przedstawia ławicę kolorowych ryb pływającą w zatopionych ruinach antycznej świątyni.

Scena składa się z wielu elementów, w większości teksturowanych proceduralnie:

-około 60 drobnych rybek, -rekin, -ruiny świątyni, -kilka roślin, -zielona rafa, -bąbelki powietrza, -piaszczyste dno.

Większość modeli pochodzi ze strony turbosquid.com

W scenie można zobaczyć 2 zaawansowane elementy graficzne: Perlin Noise i Worley Noise (więcej przy omówieniu shaderów).

Ze sceną można wejść w interakcję na kilka sposobów:

-przełączać klawiszem 'm' pomiędzy trybem automatycznego ruchu kamery i trybem manualnym

-w trybie automatycznym można zmieniać prędkość ruchu kamery klawiszami '+' i '-' (5 stopni prędkości)

-w trybie manualnym można dowolnie poruszać się kamerą: wsad, zx (jak na ćw6), dodatkowo zaimplementowałem możliwość poruszania się góra-dół za pomocą przycisków 'r' i 'f'

-pause mode: w trybie manualnym można zatrzymać i uruchomić ponownie scenę klawiszem 'p', zachowując przy tym możliwość poruszania się

Tekstury generowane są za pomocą odpowiednio dobranych 7 par shaderów, program wykorzystuje 3 vertex shadery oraz 7 fragment shaderów

Fragment shadery:

-shader\_tex: nie zmieniony w porównaniu z ćw6, wykorzystany do tworzenia przezroczystych bąbelków

-shader\_tex\_specular: j.w. + dodałem odbicie światła rozproszonego według modelu Phong'a (ćw4),

-shader\_color: nie zmieniony z ćw6

-shader\_sharp\_stripes: zwraca teksturę z ostrymi paskami w dwóch kolorach, z możliwym wyborem szerokości paska (wyk5)

-shader\_faded\_stripes: zwraca teksturę z płynną transformacją pomiędzy kolorami pasków w dwóch kolorach, z możliwym wyborem szerokości paska (wyk5)

-shader\_perlin: zwraca teksturę z wygenerowanymi czarnymi plamami, tzw. szumem gradientowym

-shader\_voro\_noise: zwraca animowaną teksturę, skrzyżowanie diagramu Woronoja i szumu Worley'a

Źródło 2 ostatnich shaderów: <https://gist.github.com/patriciogonzalezvivo/670c22f3966e662d2f83>

Każda para shaderów jest wykorzystywana w kodzie przez jedną z 7 funkcji typu drawObject\_\_\_\_\_, każda z funkcji wysyła za pomocą zmiennych typu uniform odpowiednie parametry do shaderów,

(zależnie od używanych przez dany shader): czas, kolor, szerokość pasków, kierunek światła, plik tekstury, pozycję kamery, macierze transformacji.

Bąbelki powietrza są przezroczyste dzięki aktywowaniu funkcji `GL_BLEND` w `renderScene` przed wywołaniem `drawObject___`, oraz jej deaktywowaniu zaraz po wytworzeniu obiektów. Dzięki zastosowaniu odpowiedniej macierzy translacji bąbelki poruszają się do góry, a po osiągnięciu określonego pułapu są przenoszone "pod ziemię" i ponownie zaczynają poruszać się do góry. Bąbelki są losowych rozmiarów.

Ruch kamery oraz ruchy rybek są generowane poprzez zastosowanie odpowiednich macieży transformacji, z wykorzystaniem funkcji interpolującej `catmullRom`.

Startowe pozycje ruchomych obiektów w scenie są wygenerowane przez manipulacje wektorami `vec3`, bądź też poprzez wykorzystanie funkcji `ballRand()`