

Dokumentacja programu Hawker (rozwiązanie polecenia „Programista – kulki”)

Konfiguracja programu znajduje się w pliku /Data/config.txt. Można w niej dostosować zarówno wygląd sceny jak i zachowanie ptaków i sokołów.

Program został zbudowany na szkielecie aplikacji 3D stworzonym na potrzeby mojej pracy magisterskiej. Szkielet ten dzieli kod programu na zestaw niezależnych scen. Scena odpowiedzialna za treść gry znajduje się w pliku /Source Files/SceneWorld.*. Kod tej sceny odpowiada za przetwarzanie zdarzeń wejścia (klawiatura i mysz), przygotowanie sceny do wyrenderowania świata oraz wyświetlenie nakładki HUD (celownik, energia wysłanych sokołów, FPS, ilość ptaków).

Świat może być oglądany z dwóch perspektyw:

- widok FPS – widok z oczu łowczego – sterowanie WSAD, QE (dół/góra), mysz, SHIFT (znacznie zwiększa prędkość poruszania).
- widok z kamery śledzącej wysłanego sokoła – [ENTER] lub [Prawy Klik] powraca do widoku FPS. Jeśli sokół zamieni się w ptaka, śledzony jest dalej ptak. Jeśli śledzony ptak zostanie zjedzony, śledzony jest sokół który go zjadł.

Świat oświetlony jest dwoma światłami:

- światło nieskończone (nocne lub dzienne w zależności od konfiguracji).
- światło pochodni – w trybie FPS umieszczone na pozycji łowczego, w trybie śledzącym umieszczone na pozycji śledzonego obiektu.

Logika związana z zachowaniem ptaków, sokołów, a także struktury danych opisujące stworzenia oraz miasto znajdują się w katalogu /World:

- Klasa Bird opisuje ptaka.
- Klasa Hawk opisuje sokoła (dziedziczy z klasy Bird).
- Klasa Quarter opisuje kwartał z pojedynczym budynkiem.
- Klasa Map zawiera zbiór kwartałów oraz znajdujące się w mieście stworzenia.

Za renderowanie mapy odpowiada klasa /Graphics/OGL/MapRender.*. Renderuje ona miasto w paczkach po X*X budynków, gdzie X jest rozmiarem dzielnicy określonym w pliku konfiguracyjnym. Miasto renderowane jest z wykorzystaniem shader'a GLSL /Data/shaders/IP_ADS_Bump.*. Program ten odpowiedzialny jest za oświetlenie per pixel, mapowanie wypukłości oraz zabarwianie ścian kolorem bloku (barwi wszystko poza oknem i okiennicami).

Sufit oraz ptaki renderowane są z wykorzystaniem shader'a per pixel /Data/shaders/IP_ADS_noText.*.

W przypadku braku obsługi programów GPU (lub wyłączenia ich w konfiguracji), wykorzystywane jest domyślne renderowanie OpenGL.

Program posiada również scenę konsoli (aktywacja klawiszem '~', powrót [ESC]) która wyświetla statystyki prędkości wykonania poszczególnych fragmentów kodu. Ponowne wciśnięcie klawisza '~' powoduje wyłączenie pauzy (wyświetlanie konsoli w trybie nakładki na grę).