

Opis struktury programu:

Ważniejsze struktury:

- g7_stage
 - paczka zawierająca kontekst SDL i nuklear oraz flagi
- g7_vertex
 - para liczb całkowitych
- g7_map
 - struktura reprezentująca mapę
 - g7_tile** tab
 - tablica w pamięci zawierająca mapę
 - g7_vertex size
 - rozmiary tablicy
- g7_player_state
 - stan gracza
 - g7_vertex pos
 - pozycja na mapie
 - g7_vertex vel
 - prędkość gracza
- g7_game_state
 - stan rozgrywki
 - g7_map map
 - mapa
 - int max_delta
 - maksymalna różnica prędkości w każdej osi
 - bool a_win
 - czy gracz A wygrał
 - bool b_win
 - czy gracz B wygrał
 - bool draw
 - czy remis
 - g7_player_state playerA
 - g7_player_state playerB
 - stany odpowiednich graczy
- g7_client_state
 - stan klienta (tylko jedna kopia programu)
 - bool new_own_move
 - czy wykonaliśmy ruch
 - bool new_enemy_move
 - czy wróg wykonał ruch
 - g7_player_state* own_state
 - wskaźnik na nasz stan
 - g7_player_state* enemy_state
 - wskaźnik na stan wroga
 - char buf_cmd[32]
 - ostatnia komenda wroga
 - char own_buf_cmd[32]
 - ostatnia nasza komenda
 - używane tylko przez serwer
 - float scale
 - skala mapy w oknie

- float x_offset
 - szerokość interfejsu po lewej
- TCPsocket socket
 - gniazdo SDLNet
- bool host
 - czy jesteśmy hostem

Odpowiednie moduły (dokładne komentarze w kodzie):

- connect_dialog
 - interfejs łączenia się klienta z serwerem oraz wymiana stanu gry
- end_game
 - plansza końca gry
- engine
 - ważne funkcje dotyczące rozgrywki
- gameplay
 - właściwa rozgrywka
- load_dialog
 - interfejs ładowania zapisanej gry
- main
 - punkt wejściowy gry
 - obsługa innych etapów gry
- net_utils
 - przydatne funkcje sieciowe
- new_game_dialog
 - nowa gra
- parser
 - interpretacja komend
 - czytanie stanu z pliku
- saves
 - obsługa zapisu do plików stanu gry
- utils
 - przydatne funkcje