

NHF-Farm Programozói Dokumentáció

Modulok és Függvények

Fő Modulok

- `main.c` : A program belépési pontja. Inicializálja az SDL-t, létrehozza az ablakot és a renderert, majd elindítja az eseménykezelő ciklust.
- `init.c` : Az SDL inicializálásáért, az ablak és a renderer létrehozásáért, valamint a takarításért felelős függvényeket tartalmazza.
- `grid.c` : A rács állapotának betöltéséért és kirajzolásáért felelős függvényeket tartalmazza.
- `texture.c` : Textúrák betöltéséért felelős függvényeket tartalmazza.
- `event_loop.c` : Az eseménykezelő ciklust és a felhasználói interakciókat kezeli.
- `ui.c` : A felhasználói felület elemeinek kirajzolásáért és az interakciók kezeléséért felelős függvényeket tartalmazza.

Függvények

`main.c`

- `int main(int argc, char *argv[])` : A program belépési pontja.

`init.c`

- `int initialize_SDL()` : Inicializálja az SDL könyvtárat.

- `SDL_Window *create_window(const char *title, int width, int height)` : Létrehozza az SDL ablakot.
- `SDL_Renderer *create_renderer(SDL_Window *window)` : Létrehozza az SDL renderert.
- `void cleanup(SDL_Renderer *renderer, SDL_Window *window)` : Takarítja az SDL erőforrásokat.

grid.c

- `void read_grid_state(const char *filename)` : Betölti a rács állapotát egy fájlból.
- `void render_grid(SDL_Renderer *renderer, SDL_Texture *tilemap, int tilemap_width, int tilemap_height, double zoom_level, int offset_x, int offset_y)` : Kirajzolja a rácsot.
- `void convert_to_grid_coordinates(int character_x, int character_y, int tile_size, int *grid_x, int *grid_y)` : Átkonvertálja a karakter koordinátáit rács koordinátákra.
- `void highlight_grid_square(SDL_Renderer *renderer, int grid_x, int grid_y, int tile_size, double zoom_level, int offset_x, int offset_y)` : Kiemeli a rács egy négyzetét.

texture.c

- `SDL_Texture *load_texture(SDL_Renderer *renderer, const char *file_path)` : Betölt egy textúrát a megadott fájlból.

event_loop.c

- `void event_loop(SDL_Renderer *renderer)` : Az eseménykezelő ciklus.

ui.c

- `void render_ui(SDL_Renderer *renderer)` : Kirajzolja a felhasználói felület elemeit.
- `void render_button(SDL_Renderer *renderer, ButtonType button)` : Kirajzol egy gombot a megadott típus alapján.
- `bool is_button_clicked(ButtonType button, int x, int y)` : Ellenőrzi, hogy egy adott típusú gombra kattintottak-e.

Függvények Paraméterei és Visszatérési Értékei

`initialize_SDL()`

- **Paraméterek:** Nincs
- **Visszatérési érték:** `int` - 0, ha sikeres, egyébként negatív érték hibakód esetén.

`create_window(const char *title, int width, int height)`

- **Paraméterek:**
 - `title` : Az ablak címe.
 - `width` : Az ablak szélessége.
 - `height` : Az ablak magassága.
- **Visszatérési érték:** `SDL_Window*` - Az ablak mutatója, vagy NULL hiba esetén.

`create_renderer(SDL_Window *window)`

- **Paraméterek:**
 - `window` : Az ablak mutatója, amelyhez a renderert létrehozzuk.

- **Visszatérési érték:** `SDL_Renderer*` - A renderer mutatója, vagy NULL hiba esetén.

```
cleanup(SDL_Renderer *renderer, SDL_Window *window)
```

- **Paraméterek:**
 - `renderer` : A renderer mutatója.
 - `window` : Az ablak mutatója.
- **Visszatérési érték:** Nincs

```
read_grid_state(const char *filename)
```

- **Paraméterek:**
 - `filename` : A fájl neve, amelyből a rács állapotát betöltjük.
- **Visszatérési érték:** Nincs

```
render_grid(SDL_Renderer *renderer, SDL_Texture *tilemap,  
int tilemap_width, int tilemap_height, double zoom_level,  
int offset_x, int offset_y)
```

- **Paraméterek:**
 - `renderer` : A renderer mutatója.
 - `tilemap` : A tilemap textúra mutatója.
 - `tilemap_width` : A tilemap szélessége.
 - `tilemap_height` : A tilemap magassága.
 - `zoom_level` : A nagyítás mértéke.
 - `offset_x` : Az x tengelyen eltolás.
 - `offset_y` : Az y tengelyen eltolás.
- **Visszatérési érték:** Nincs

```
convert_to_grid_coordinates(int character_x, int
character_y, int tile_size, int *grid_x, int *grid_y)
```

- **Paraméterek:**

- `character_x` : A karakter x koordinátája.
- `character_y` : A karakter y koordinátája.
- `tile_size` : A rács mérete.
- `grid_x` : A rács x koordinátája.
- `grid_y` : A rács y koordinátája.

- **Visszatérési érték:** Nincs

```
highlight_grid_square(SDL_Renderer *renderer, int grid_x,
int grid_y, int tile_size, double zoom_level, int
offset_x, int offset_y)
```

- **Paraméterek:**

- `renderer` : A renderer mutatója.
- `grid_x` : A négyzet x koordinátája a rácsban.
- `grid_y` : A négyzet y koordinátája a rácsban.
- `tile_size` : A négyzet mérete.
- `zoom_level` : A nagyítás mértéke.
- `offset_x` : Az x tengelyen eltolás.
- `offset_y` : Az y tengelyen eltolás.

- **Visszatérési érték:** Nincs

```
load_texture(SDL_Renderer *renderer, const char
*file_path)
```

- **Paraméterek:**

- `renderer` : A renderer mutatója.

- `file_path` : A fájl útvonala.
- Visszatérési érték: `SDL_Texture*` - A textúra mutatója, vagy NULL hiba esetén.

`event_loop(SDL_Renderer *renderer)`

- Paraméterek:
 - `renderer` : A renderer mutatója.
- Visszatérési érték: Nincs

`render_ui(SDL_Renderer *renderer)`

- Paraméterek:
 - `renderer` : A renderer mutatója.
- Visszatérési érték: Nincs

`render_button(SDL_Renderer *renderer, ButtonType button)`

- Paraméterek:
 - `renderer` : A renderer mutatója.
 - `button` : A gomb típusa.
- Visszatérési érték: Nincs

`is_button_clicked(ButtonType button, int x, int y)`

- Paraméterek:
 - `button` : A gomb típusa.
 - `x` : Az egér x koordinátája.
 - `y` : Az egér y koordinátája.
- Visszatérési érték: `bool` - Igaz, ha a gombra kattintottak, egyébként hamis.