**Game Programming Resources**

C and SDL2

**Learn Video Game Programming in C** https://www.youtube.com/playlist?list=PLT6WFYYZE6uLMcPGS3qfpYm7T\_gViYMMt

Notes

• To maintain the complexity of the main function, event handling and rendering is done in separate functions called ‘processEvents()’ and ‘doRender()’ respectively. this frees up the main application loop.

• To do keyboard event handling, the SDL\_GetKeyboardState() function is used instead of the event.key.keysym.sym enumeration. The SDL\_GetKeyboardState() works similarly to the SDL\_GetMouseState() function, but instead of returning information about the keyboard as output parameters (there are way to many keyboard states to do that), this function returns a pointer to an array of key states.

• Create one object that encapsulates all of the game data (called Gamestate). The root level hierarchical structure. This includes resources for the game.

typedef struct {

// Players Man man;

// Images SDL\_Texture\* star; } GameState;

• Use a separate function to do all of the initialization of game state data and resource loading. called ‘loadGame()’.

• An interesting technique that the author uses is to use a single generic surface to create textures, instead of creating a new surface for each texture:

SDL\_Surface\* surface;

// Load first image surface = IMG\_Load(“/images/image1”); image1 = SDL\_CreateTextureFromSurface(renderer, surface); SDL\_FreeSurface(surface);

// Load second image surface = IMG\_Load(“/images/image2”); image2 = SDL\_CreateTextureFromSurface(renderer, surface); SDL\_FreeSurface(surface);

**Learning C Game Programming - Simple Gun Shooting** https://www.youtube.com/watch?v=FCRmIoX6PTA

**Learning C Game Programming - Simple Particles** https://www.youtube.com/watch?v=xYHkJQKBf8g

**Tic-Tac-Toe Game in C with SDL — Procedural vs Functional** https://www.youtube.com/watch?v=gCVMkKgs3uQ

Tile-Based Game Programming

**SDL2 Isometric Game Tutorial** https://www.youtube.com/watch?v=8xf\_ouTpd1c

**Creating Isometric Worlds: A Primer for Game Developers** https://gamedevelopment.tutsplus.com/tutorials/creating-isometric-worlds-a-primer-for-game- developers--gamedev-6511

**An Updated Primer for Creating Isometric Worlds, Part 1** https://gamedevelopment.tutsplus.com/tutorials/creating-isometric-worlds-primer-for-game- developers-updated--cms-28392

**Create Your Own Isometric Tile-Based Game: Part 1** http://bigspritegames.com/isometric-tile-based-game-part-1/