CS 250

Intro to Game Programming with SDL

Spring 2016
Topics

- API
- SDL
- Screen Coordinates
- Game Loop
- Simple Animation
API

- Application Program Interface (API)
- API
  - describes an interface of a software system
  - is a set of commands, functions, and protocols programmers can use when building software
  - allows programmers to use predefined functions when developing software
  - examples include: Standard Template Library in C++, Java API, SDL, Google Maps, Android
SDL

- SDL is an API that can be used with C++, C#, Python, and other languages, to write games
- SDL is available on Windows, Mac, Linux, iOS, and Android
- [https://www.libsdl.org/](https://www.libsdl.org/)
Object Oriented Design

• SDL is not an Object Oriented API
  – A set of functions is provided
  – Procedural programming

• We will build objects that use these functions
• We will build up a set of useful objects to help you later build a 2D game
• No objects today, however.
SDL in Visual Studio

• Visual Studio uses the acronym SDL to mean:
  – Security Development Lifecycle
  – Command line option
    /sdl
SDL Setup
for your home computer

- SDL Setup folder is in CS250 Public
- README Instructions exist in the folder
- The setup is for Visual Studio 2013
- I have only tested this setup on Windows 7 systems
- SDL machines in the CS Lab, Marsh Labs, and 24-hour library lab
SDL Software

• https://www.libsdl.org/download-2.0.php

Development libraries
http://wiki.libsdl.org/MigrationGuide#Add-on_libraries

• https://www.libsdl.org/projects/SDL_ttf/
• https://www.libsdl.org/projects/SDL_mixer/
• https://www.libsdl.org/projects/SDL_image/
Setup your project to use SDL

- Project | Properties | Configuration Properties
  - VC++ Directories
    - Include Directories:
    - c:\Program Files (x86)\SDL2.0.3\include
  - Library Directories
    - c:\program files (x86)\SDL2.0.3\lib\x86

- Linker | Input
  - Additional Dependencies:
  - SDL2.lib;SDL2main.lib;SDL2_gfx.lib;SDL2_ttf.lib
Setup your project
HelloSDL

Copy HelloSDL from Grace to your desktop

Compile and run

Let’s discuss what is happening.
SDL pieces

- Window
- Renderer
- Texture
- Surface
Screen Info

- Upper left is (0, 0)
- RGBA or a 32-bit value storing the red channel value (8-bits), green (8-bits), blue (8-bits), and alpha channel (8-bits) used for transparency
- Alpha channel values
  - 0 is fully transparent
  - 255 is fully opaque
SDL_SimpleGraphics

#include "SDL2_gfxPrimitives.h" // primitive graphics

// Change color to blue!
SDL_SetRenderDrawColor (pRenderer, 0, 0, 255, 255);

// Render a circle
filledCircleRGBA (pRenderer, 100, 100, 50, 0, 0, 255, 255);
SDL Documentation

- [http://wiki.libsdl.org/FrontPage](http://wiki.libsdl.org/FrontPage)

Google search for "sdl color"
Color Example

• You can use a Color Picker to help you select the appropriate color:
  – Use the R G B numbers at the bottom
Problem

• Using the help functions, edit the program SDL_SimpleGraphics to draw 5 circles in a random location on the screen.

• Challenge: Randomize the color of the circles as well.
SDL Simple Animation

- SDL Simple Animation
- Render a circle
- Clear screen
- Render again

- Frame Rate
Loop Animation

LOOP

SDL_RenderClear

//draw scene
SDL_RenderPresent

//Force Frame Rate
SDL_SimpleAnimation

• What happens if we remove SDL_RenderClear from the code?

• Modify the code on the previous slide to animate two circles.
  – One starts from the top left and moves down to the bottom right,
  – and the other starts from the top right and moves down to the bottom left
SDL_SimpleAnimation

- Modify the code so that the radius of the circles starts at 1 and grows by 1 during every iteration of the game loop.

- Modify the code so that the radius of the circle starts at 1 and grows by 1 for every 5th iteration of the game loop.
Object Oriented Design

- Let’s start to build Objects that use the SDL API
- What functionality/data does a Circle have?
Composition

• Build a class Snake that is made up of 4 circles, each of a different color.

• As the Snake moves, all 4 circles should move correctly.