

**Department of Computer Science, Clemson University**  
**CpSc 416/616 – 2D Game Engine Construction with C++**  
**Topics Covered: November 11, 2008**

First Exam Topics:

1. Drawing a surface with SDL
  2. BLock Image Transfer (blit); KyleXY
  3. Colorkeys and transparency
  4. Loading image formats
  5. OO: The Basic Game Framework
  6. Sprite animation
  7. Sprite animation with frames
  8. Parallax Scrolling
  9. Processing user input: keyboard
  10. SDL Timer: Game time and timers
  11. Adding text with True Type Font
  12. Using a singleton for IO: IOSingleton
  13. Computing frames per second
  14. Storing sprites in STL Vectors
- 

Second Exam Topics

1. Using vectors & lists: erase
2. Finding Leaks with Valgrind
3. Incorporating music and sound
4. Collision detection
5. Vectors: Shooting
6. Explosions: chunking
7. Using rotozoom for fx
8. Inheritance: virtual, non-virtual, pure

9. Polymorphism
  10. Static & dynamic casting
  11. The composite pattern: levels
- 

Remaining Topics

1. Pausing the game
  2. Saved games
  3. Winning & Losing
  4. Mouse/Controller
  5. Networking
  6. Artificial Intelligence
- 

C++ Topics

1. **Basics** (100 slides, 1-2 classes): I/O, data types, iteration: for, while; control structures: if/else and switch; short-circuit evaluation, functions, The 3 parameter transmission modes, C-strings, C++ strings, command-line parameters, intro to files. String streams.
2. **The C++ Class** (100 slides, 6-8 classes): Constructors: default, conversion and copy. Class instantiation: constructor initialization vs assignment, when are constructors called (which one!), destructors and when they are required. The class members that are supplied by default, those that you should supply. Shallow vs deep copy, orthodox canonical class form, functions that C++ silently writes, overloading functions, overloading *operators*, overloadly assignment, the output operator and others. friend functions (functions that are members of a class vs part of a class), nomenclature and programming style, make files, dynamic

vs static storage, Writing a string class. Deep copy vs shallow copy. Dynamic memory allocation. Comparing C string with C<sup>++</sup> strings. the stack class, the template stack class, exceptions.

3. **The standard library:** (43 slides, 2-4 classes) `vector`.
4. **Inheritance:** (43 slides, 2-4 classes) what are the kinds of functions that can be involved in an inheritance relation (virtual, purely virtual, non-virtual). When should inheritance be used, what are the problems with inheritance.
5. **Design Patterns:** Singleton.