

Midterm Exam Review Topics

C++ Basics

1. Properties of C++.
2. C++ file types.
3. Stages of a program creation. Kinds of errors that can be discovered at each stage. Compilation and linking. What does each do? What files are needed? What files are produced?
4. C++ types: primitive (built-in) value types, `class` types, `enum` types, `typedef`, reference types, and pointer types.

I/O

1. Streams: `cin`, `cout`, `cerr`, and `clog`.
2. Handling files.
3. Manipulators.
4. Kinds of I/O errors and ways to handle them.
5. Error flags, their meaning and how they are set.
6. Extending operators. What operators can be extend and how? Advantages of extending operators.

Classes and objects

1. The roles of a class.
2. Class elements and class structure.
3. Qualifiers: `inline`, `const`, `static`.
4. Class member visibility: `private`, `protected`, `public`.
5. Differences between a class declaration and a class implementation. Importance of a class interface.
6. Constructing/destructing objects.
7. Different kinds and roles of constructors: ctors, copy constructors, default constructors, null constructor.
8. Initialization, assignment, copying of objects.

Functions and methods

1. Passing data to a function. What's are the different ways to pass data to a function? Advantages of each method.
2. Receiving data from a function. How data can be passed back from a function? Can more than one value be returned?
3. What is the implicit argument to a function? What kinds of functions have implicit arguments?
4. Default parameters.

Pointers and references

1. L-values and R-values.
2. References vs. pointers.
3. Pointer R-values.
4. Notation for reference and pointer types.
5. Operators `&` and `*`.
6. How is an array like a pointer, and how does it differ?

Storage model

1. Storage classes: static, stack, heap.
2. `new` and `delete`.
3. Object lifetime.
4. Object size, `sizeof`.
5. Dangling pointers and their causes: failure to initialize, premature deletion of target.
6. Five kinds of failures. When different errors can occur?
7. Memory leaks - what they are, how they arise, how they can be detected and design patterns for avoiding them.
8. Memory management paradigms.
9. Difference between `new` and `new []`, `delete` and `delete []`.

C++ Standard Library

1. What is the C++ Standard Library and how can it be used?
2. Three classes: `string`, `stringstream` and `vector`.

Class dependencies

1. Tightly coupled classes.
2. Circular dependencies.
3. Introduction to UML (Unified Modeling Language).
4. Elementary design principles.

(Thanks to Ewa Syta for an earlier draft of these notes.)