On to C++

I made up the term "object-oriented", and I can tell you I did not have C++ in mind. — Alan Kay

It does a lot of things half well and it's just garbage heap of ideas that are mutually exclusive. — Ken Thompson

Inside C++ is a smaller, cleaner, and even more powerful language struggling to get out. And no, that language is not C, C#, D, Haskell, Java, ML, Lisp, Scala, Smalltalk, or whatever.
— Bjarne Stroustrup

Stroustrup’s overview of C++:
Homework: read this paper.
Object-orientation in Java

class Animal {
    public void speak() {
        System.out.println("Generic animal noise");
    }
}

class Pig extends Animal {
    public void speak() {
        System.out.println("Grunt!");
    }
}

class Pigtest {
    public static void main(String[] args) {
        Animal peppa = new Pig();
        peppa.speak();
        (new Animal()).speak();
    }
}
Object-orientation in C++

class Animal {
    public: virtual void speak() {
        cout << "Generic animal noise\n";
    }
};

class Pig : public Animal {
    public: void speak() {
        cout << "Grunt!\n";
    }
};

int main(int argc, char* argv[]) {
    Animal *peppa = new Pig();
    peppa->speak();
    (new Animal())->speak();
}
C++ compared to C

- C++ is more modern and high-level than C
- C++ has abstraction mechanisms: OO and templates
- In C++, classes are like structs.
- Fundamental design decision: OO is spatchcocked into C.
- By contrast, in Objective-C, objects are a layer on top of C separate from struct.
- Arguably, Objective-C is more object-oriented than C++ and Java
- C is a simple language, C++ is extremely complicated
C++ compared to Java

- Java does not contain C, C++ does
- C++ is more fine-grained than Java.
- Java vs C++
  - Java inheritance: methods = virtual functions and public inheritance, not implementation-only
  - Java new is garbage-collected
  - C++ new is like a typed malloc, must use delete
- Constructors and destructors in C++
We will only use a Java-like subset of C++ OO system

- Only single inheritance
- Only virtual member functions
- Heap-allocated objects
- No multiple or private inheritance
- If you don’t understand some part of C++, don’t use it
- We use a subset similar to Java type system
- Even so: need memory management: destructors and delete
- printf did not need fixing. Stroustrup fixed it anyway. << is overloaded with IO operations.
What C/C++ is like
What C/C++ is like
What C/C++ is like

"THIS IS SPARTA"
What C/C++ is like

Pointers and recursion
What C/C++ is like

Pointers and recursion and types
And what is full of pointers and recursion and types?

\[1\]But be wary of “paradigms”, especially when they “shift”. The word can mean anything from marketing guff to pseudoscience.
Trees in C/C++

And what is full of pointers and recursion and types? Parse trees are.
We can implement them with structs in C or OO in C++. These two styles of trees are important independently of C++. They represent different programming “paradigms”.¹

Lab:

¹But be wary of “paradigms”, especially when they “shift”. The word can mean anything from marketing guff to pseudoscience.