What type of language ?

C was a procedural/imperative language

("do this and then do that")

With Functional Decomposition as main paradigm:

Functions call sub-funcs that call sub-sub-funcs

Java is also a procedural/imperative language

But its (intended) paradigm is Object Orientation:

Classes and Objects organise "functions" and data

Functional Decomposition (what C is)

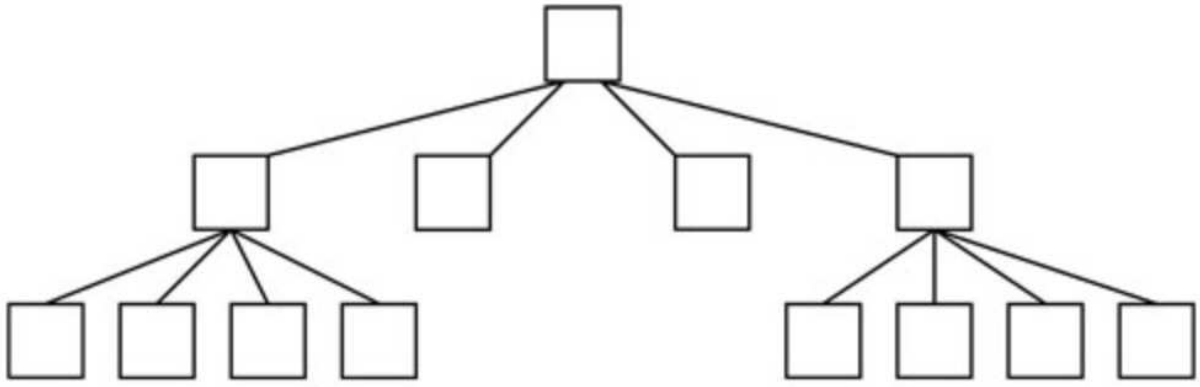
Main function is broken down into smaller functions

Forms a tree, with data flowing around everywhere

Although effective programs can be written this way

They tend to be monolithic (big and frightening)

Also "brittle" (resistant to long-term evolution)



Object Orientation

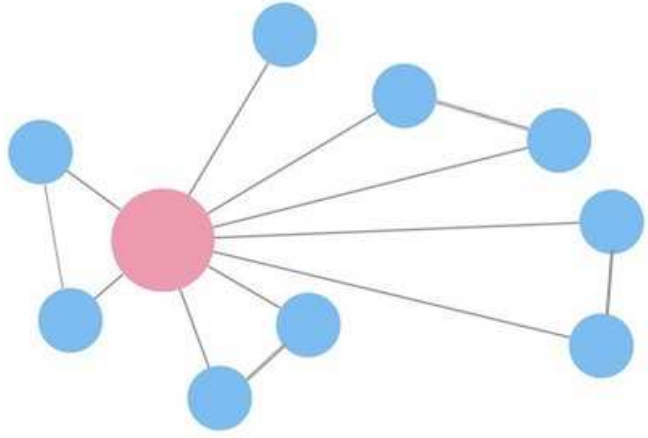
Program written as decentralised cooperating pieces

Each piece (Object) looks after its own internal data

Collaborate with each other to get the job done

Scales well to larger projects (if done properly !)

Works effectively for big teams (if done properly !)



! WARNING !

Lots of concepts and terminology
Are about to come your way !

You probably won't grasp them first time around
You will need to review these notes again
(And maybe then they will start to stick)

What are Classes and Objects?

We've mentioned them already,
but what exactly ARE Classes and Objects ?

CLASSES: modules that divid up the source code
(Normally each file contains just a single Class)

OBJECTS: structures that divid up running programs
(Each Object encloses its own state and data)

Classes can be viewed as a template (cookie cutter)
from which we can "instantiate" live Objects

Key Characteristics of Object Orientation

- Abstraction: sophistication, but with simple interface
- Encapsulation: complexity hidden "under the hood"
- Inheritance: hierarchies (like family trees) of Classes
- Polymorphism: like Classes can be treated similarly

Perhaps a bit more detail ?