

## 8 - control flow

usual control flow = a command followed by the other, executed in sequence

↳ single entrance - single exit

sequencers = commands to change control flow (makes multiple exits available)

↳ jumps

↳ escapes

↳ exceptions

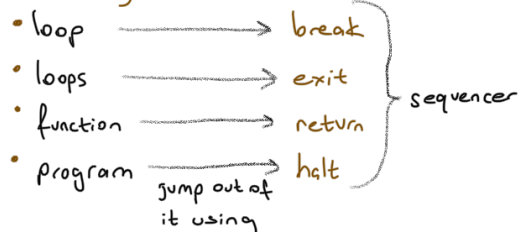
jumps: transfer control to a point in the code

↳ destination is marked with labels, like goto

spaghetti coding = when jumps to arbitrary positions are possible (unrestricted jumps)

escapes: restricted jumps to out of textually enclosing blocks.

enclosing blocks



non-local jumps = jumps out of a function block

↳ possible only for one direction: if the stack position can be recovered

exceptions: non-local, controlled jumps out of multiple levels of function calls to an outer control point → handler, catch

↳ try-catch blocks: try {} is executed when throw expr command is called

co-routines = generalize subroutines for multitasking, by allowing execution to be suspended and resumed

• non-local jumps to different environments guided coordinated programs or a global scheduling mechanism

↳ uses: callbacks, generators (iterators), threads, fibres, asynchronous, event based, or concurrent programming