

Projects

Group 1

1. MLC testing system

- hand out test with barcode
- student completes test with scanner
- grade test - attach to test
- student can view grade online, teacher can change grade online

2. online testing for MLC

3. media player for Linux

Group 2

1. grade checker and calculator

- curve calculator
- statistics

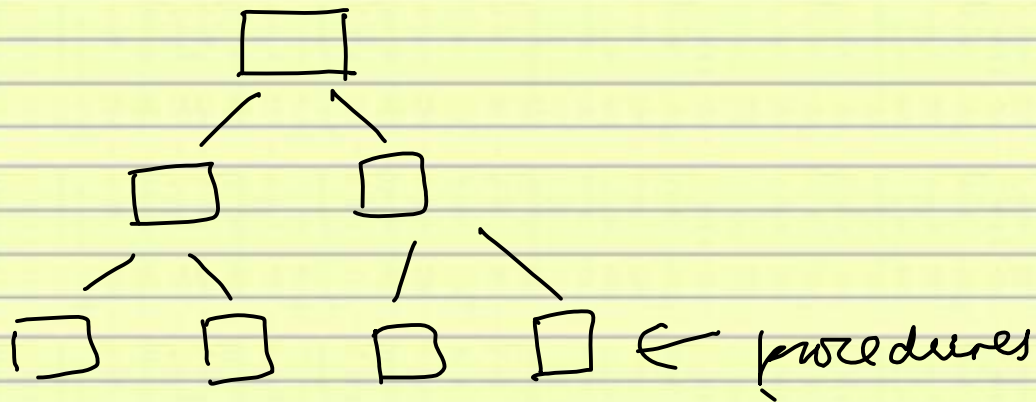
2. work study scheduler

- online
- alternatives
- multiple students

Object-oriented methodology

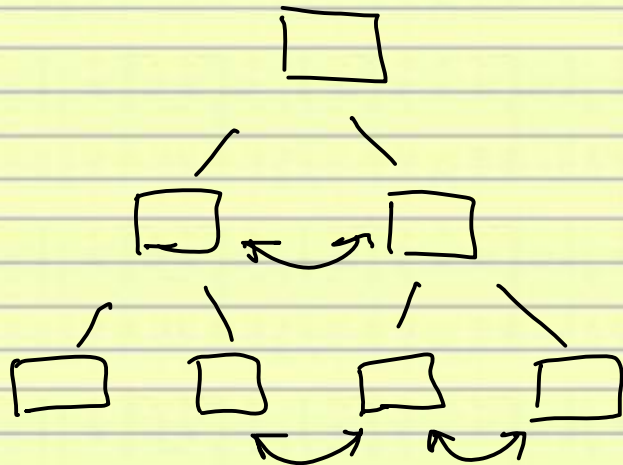
Before this we had procedural abstraction.

- a task is broken into subtasks and sub-subtasks as necessary. Top-down methodology. Each "leaf" task gets its own procedure. Sometimes the intermediate nodes can also be procedures.



Each procedure is self-contained, but data can be shared either through non-local references, or through parameters.

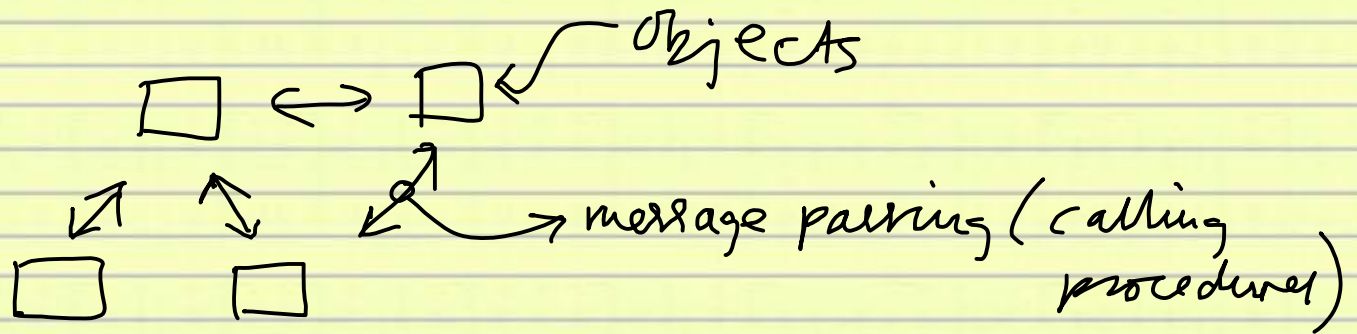
But with increasing complexity, there is too much interaction between the various nodes in the tree.



The basic tree structure breaks down.

Pascal was good at tree structures, because it allowed nested procedures.

With O-O methodology many more program structures could be used. - data abstraction.



network model of interacting objects rather than a hierarchy of procedures

Both Java and C++ (C#) support this network model:

We will use a language-independent way of talking about objects and their interactions.

UML (universal modeling language)

UML was designed in 90's and now a standard description language for SE.

UML is diagrammatic, not textual

has the following types of diagram

- ✓ 1. class diagram
 - ✓ 2. use case diagram
 - ✓ 3. package diagram - program structure
 - (✓) 4. activity
 - (✓) 5. sequence
 - (✓) 6. state
 - X 7. collaboration
 - X 8. deployment - interactions with standard sub-system
- } data and actors that use data
- } dynamic behavior (procedures)

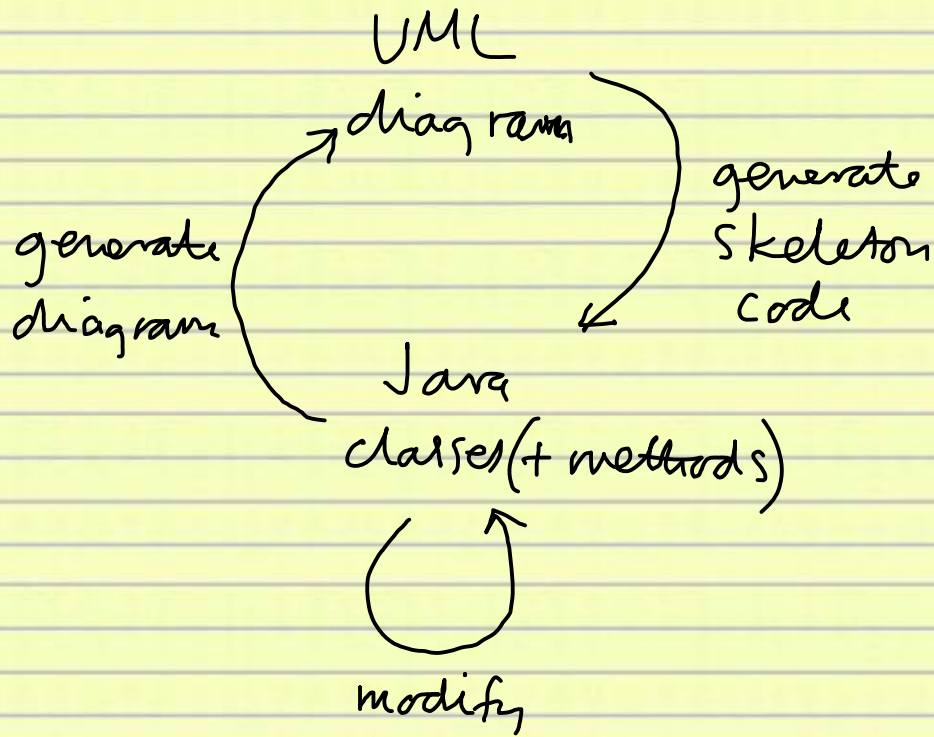
We will use class and use case diagrams to capture (describe) functional requirements.

We can use the other diagrams during program design.

UML is often used as follows;

1. conceptual modeling - diagrams have little detail
2. specify functional requirements - diagrams have some detail
3. design - diagrams have complete detail

Jude (Community) - can enable simple drawing of diagrams - can also round-trip engineering



round-trip engineering