

# Symbol Tables

**CS 4447 / CS 9545 – Stephen M. Watt**  
***The University of Western Ontario***

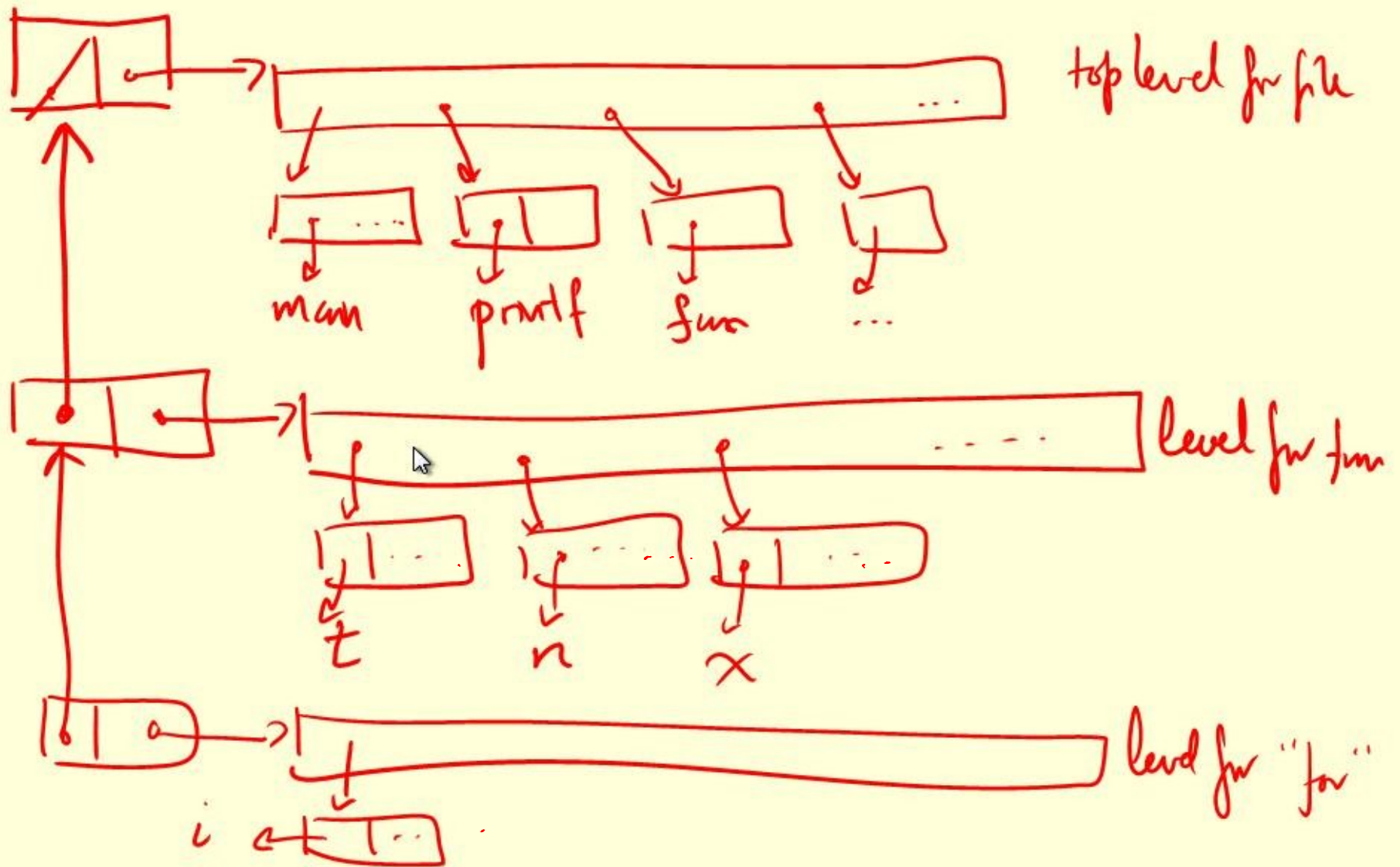
# Symbol Tables

- A data structure used to keep track of identifiers that occur in the source program.
- Used to keep track of type information, location of definition, whether used, type *etc.*
- Used for
  - variable names,
  - function names,
  - enumeration names,
  - *typedefed* names,
  - structure/class/union fields
  - Information needed for code generation
  - ...

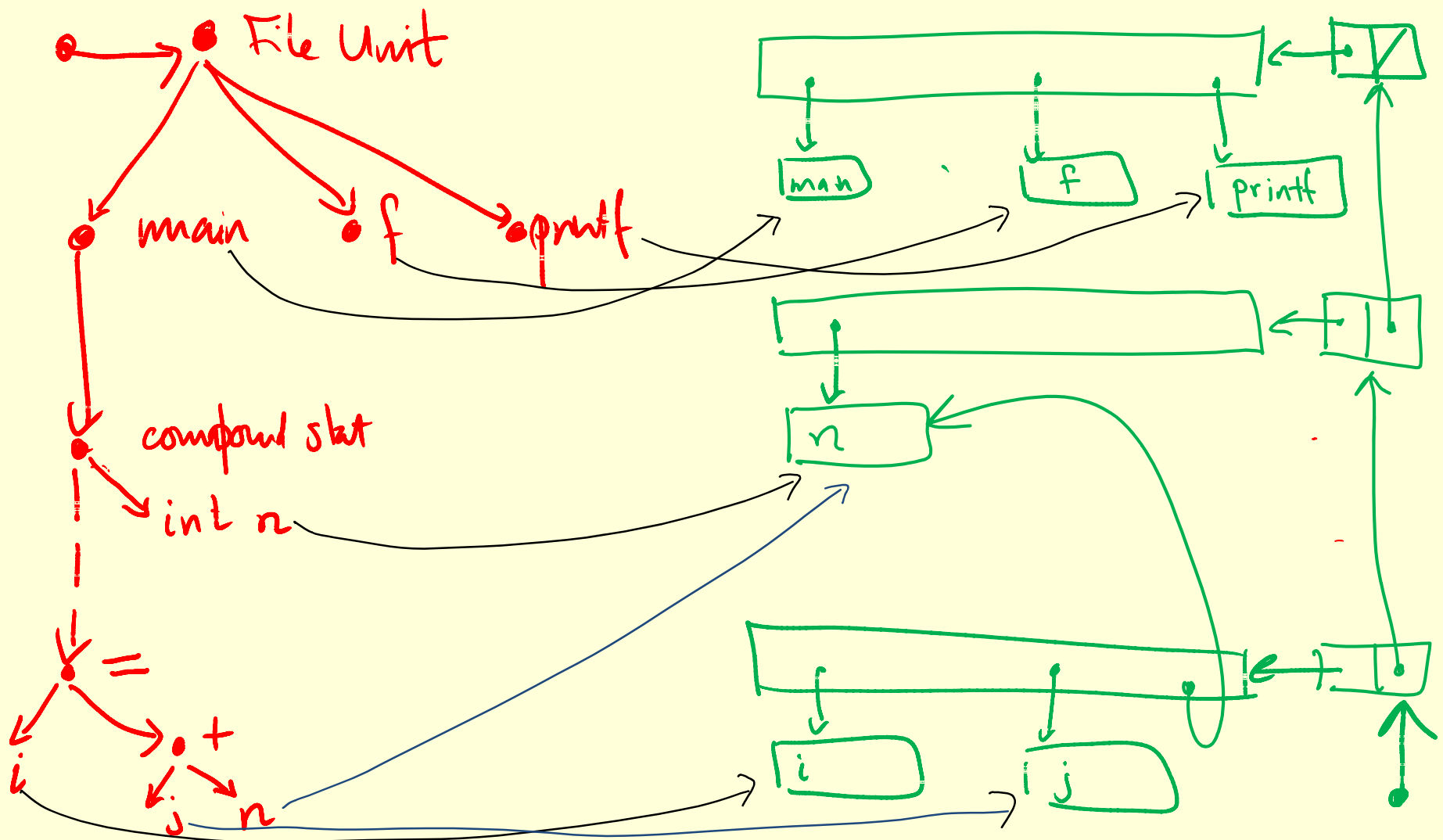
# How Used

- Structured to have tiers of levels, corresponding to nested blocks.
- Built in one or more passes during a “semantic analysis” phase.
- Symbols are added when *declarations* are encountered, and looked up on each *use*.
- The rules of the programming language govern what are allowed re-definitions, over-ridings, definition after use, etc.
- Typically have an “attribute” slot in structures representing block statements and use this to point to a symbol table level.
- Likewise have attributes on identifiers pointing to symbols.

# Symbol Table Levels



# Links to Abstract Syntax Tree



# Useful Information for Each Symbol

- For semantic analysis:
  - Level defined. Position defined.
  - Has it been used yet?
  - If so, in which levels? At what positions?  
(For error messages.)
  - Type
- For code generation:
  - Level defined (again).
  - Offset in structure/object/activation record.