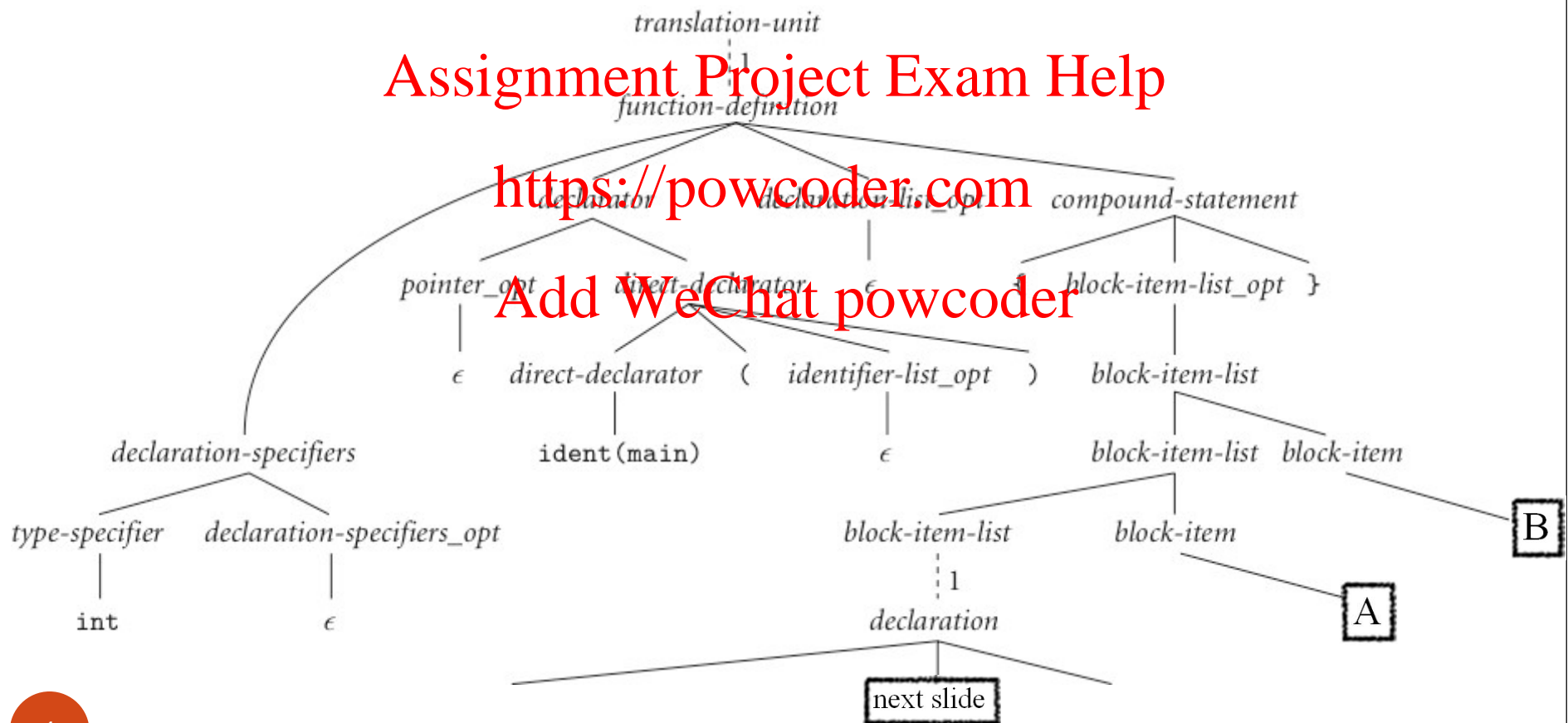


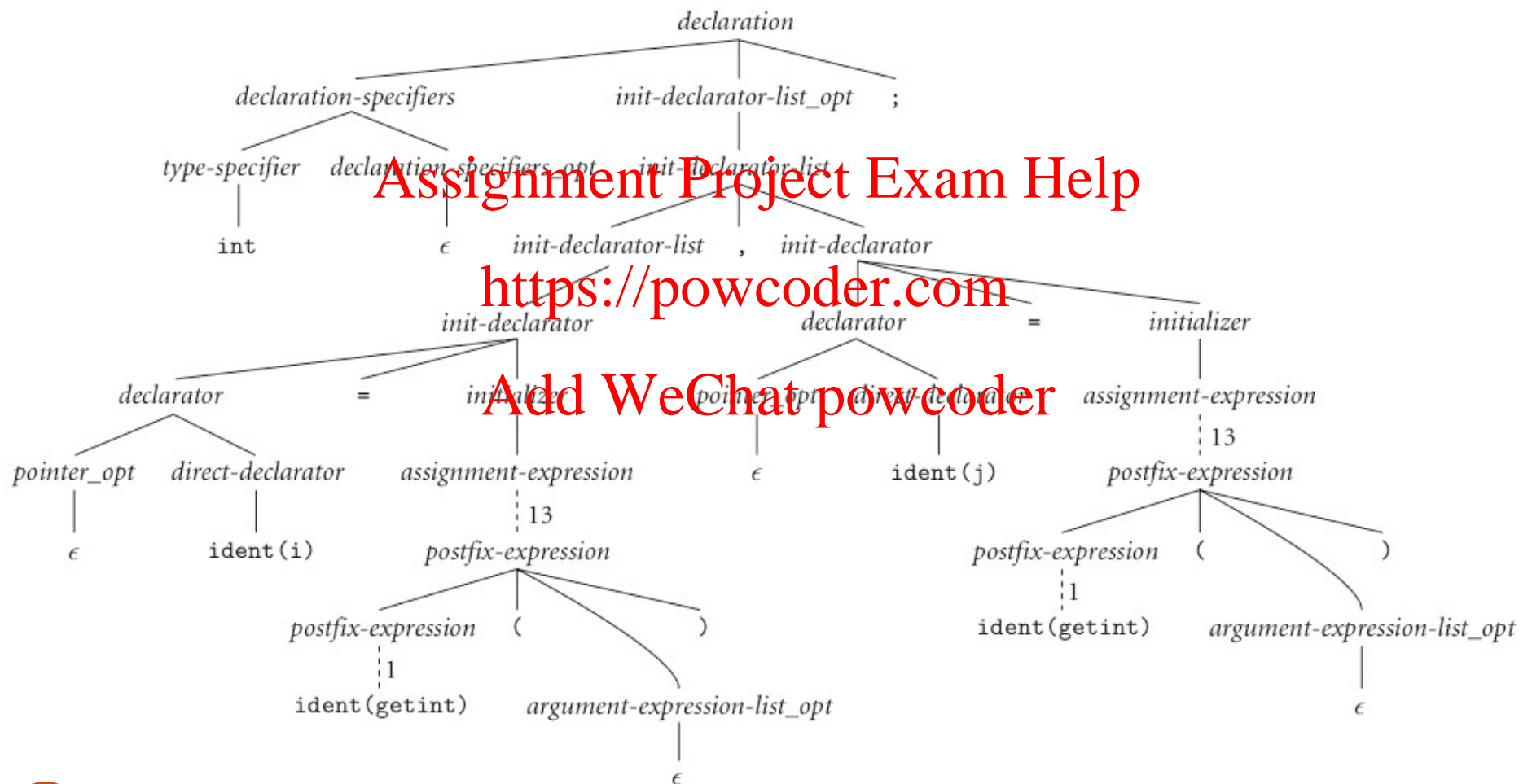
An Overview of Compilation

- Context-Free Grammar and Parsing
 - GCD Program Parse Tree:



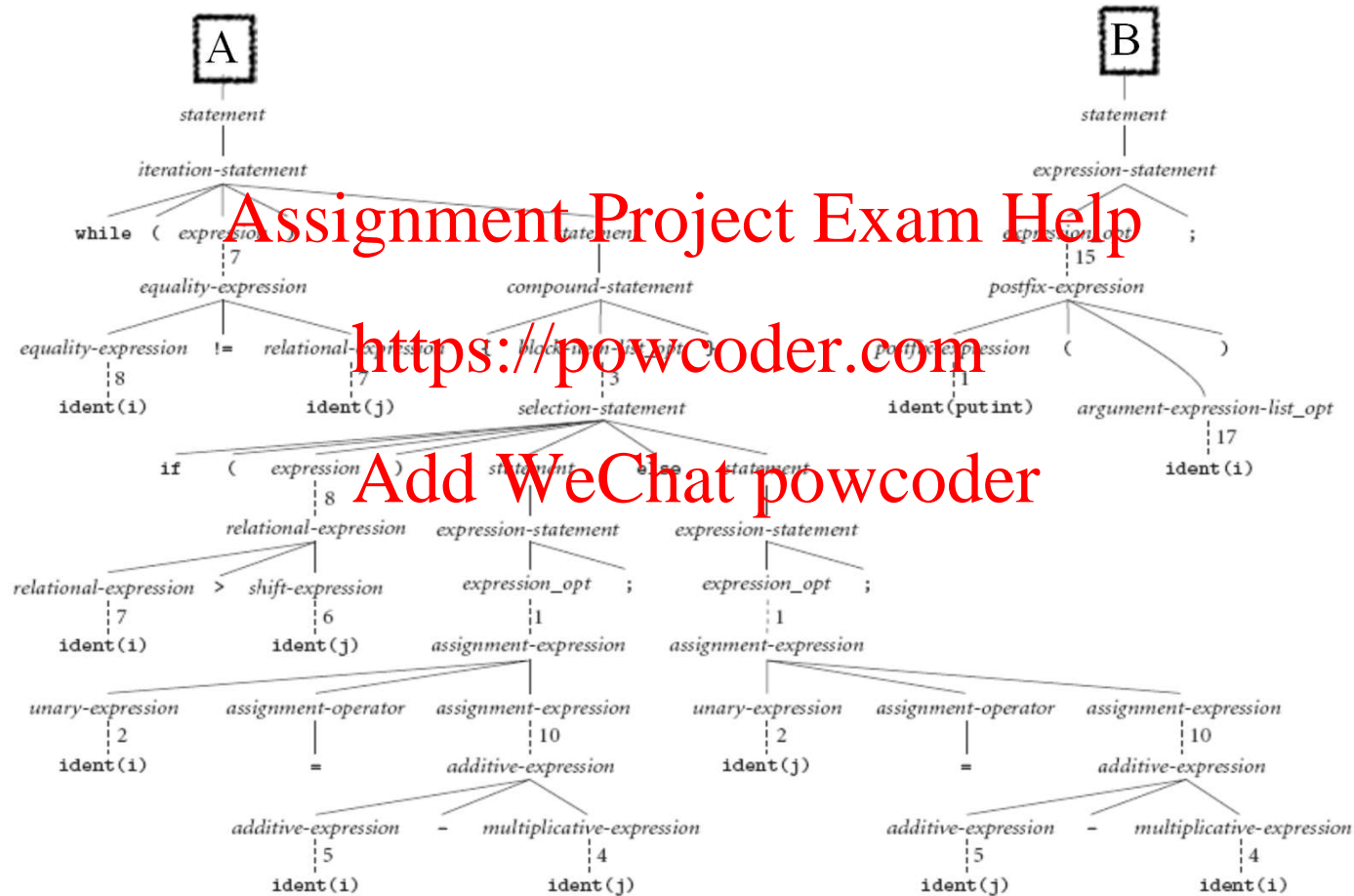
An Overview of Compilation

- Context-Free Grammar and Parsing (continued)



An Overview of Compilation

- Context-Free Grammar and Parsing (continued)



An Overview of Compilation

- Semantic Analysis and Intermediate Code Generation
 - **The parse tree is very verbose:** once we know that a token sequence is valid, **much of the information in the parse tree is irrelevant to further phases of compilation**
<https://powcoder.com>
[Add WeChat powcoder](#)
 - The semantic analyzer typically transforms the parse tree into an abstract syntax tree (AST or simply a *syntax tree*) by removing most of the “artificial” nodes in the tree’s interior
 - The semantic analyzer also *annotates* the remaining nodes with useful information, such as pointers from identifiers to their symbol table entries
 - The annotations attached to a particular node are known as its

An Overview of Compilation

- GCD Syntax Tree (AST)

