

Concepts of Programming Languages, CSCI 305, Fall 2021
Scope Rules, Nov. 10

Scope

Scope – the textual region of a program in which a name-to-object binding is active

Static scope – bindings are defined by the physical (lexical) structure of the program, thus bindings can be determined at compile time

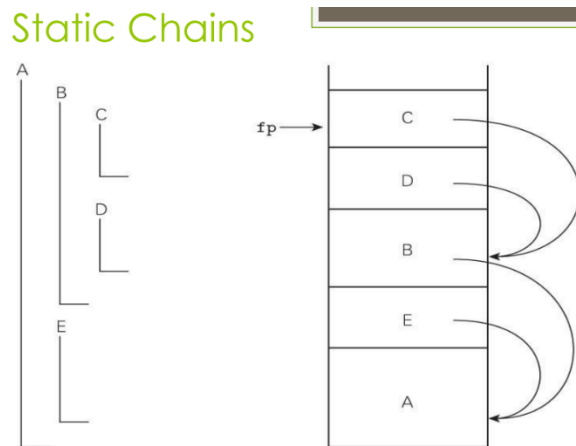
Dynamic scope – bindings depend on the current state of program execution

When subroutines are nested they can access the local variables and other local functions in the enclosing function.

Languages allowing nesting of subroutines:

- Allowed – C#, Pascal, Modula, Ada, most scripting languages (Perl, JavaScript, Python) and most functional languages (Lisp, Scheme, Ocaml)
- Not allowed - C, C++, Java

Part of the “miscellaneous bookkeeping” are chains (static or dynamic) to implement scope



Static Scope	Dynamic Scope
<ul style="list-style-type: none"> • Easier to understand • More efficient 	<ul style="list-style-type: none"> • Easier to implement • Facilitates customization of subroutines

Implementing Scope

Static scope – compiler uses symbol table

Compiler symbol table operations:

- insert new name-to-object binding

- lookup

Dynamic scope – interpreter performs operations analogous to symbol table insert and lookup at run time. Can use a list of name/value pairs and acts like a stack – push when new declarations are encountered, pop when at the end of their scope.