Introduction to SimpleSem (C3)

CS181: Programming Languages

Topics:

- Features of C3
- Activation records (AR)
- C3 return value
- Calling sequence
- Return sequence

SimpleSem overview

- C1 – a language with only simple statements
- C2 – adds simple routines to C1

- C1, C2 – static
- C3, C4 – automatic (stack-based)
- C5 – dynamic

Features of C3

- C3 adds new features to C2:

  - direct recursion

  - indirect recursion

  - return values for functions

Features of C3

- The activation record of each function has a known fixed size: 2 + number of locals
- The number of instances of any function (except main) is not known
- All different calls of the same function have the same code segment
- But we need different activation records to store the different values of the local variables

Features of C3

- It is not possible to bind each variable to its offset in the corresponding activation record until execution time.
- D[0] is the *base address* of the currently executing unit. We call it *CURRENT*
- D[1] has the address of the first *FREE* cell
- After a function completes its current instance, it is possible to free the space occupied by the activation record

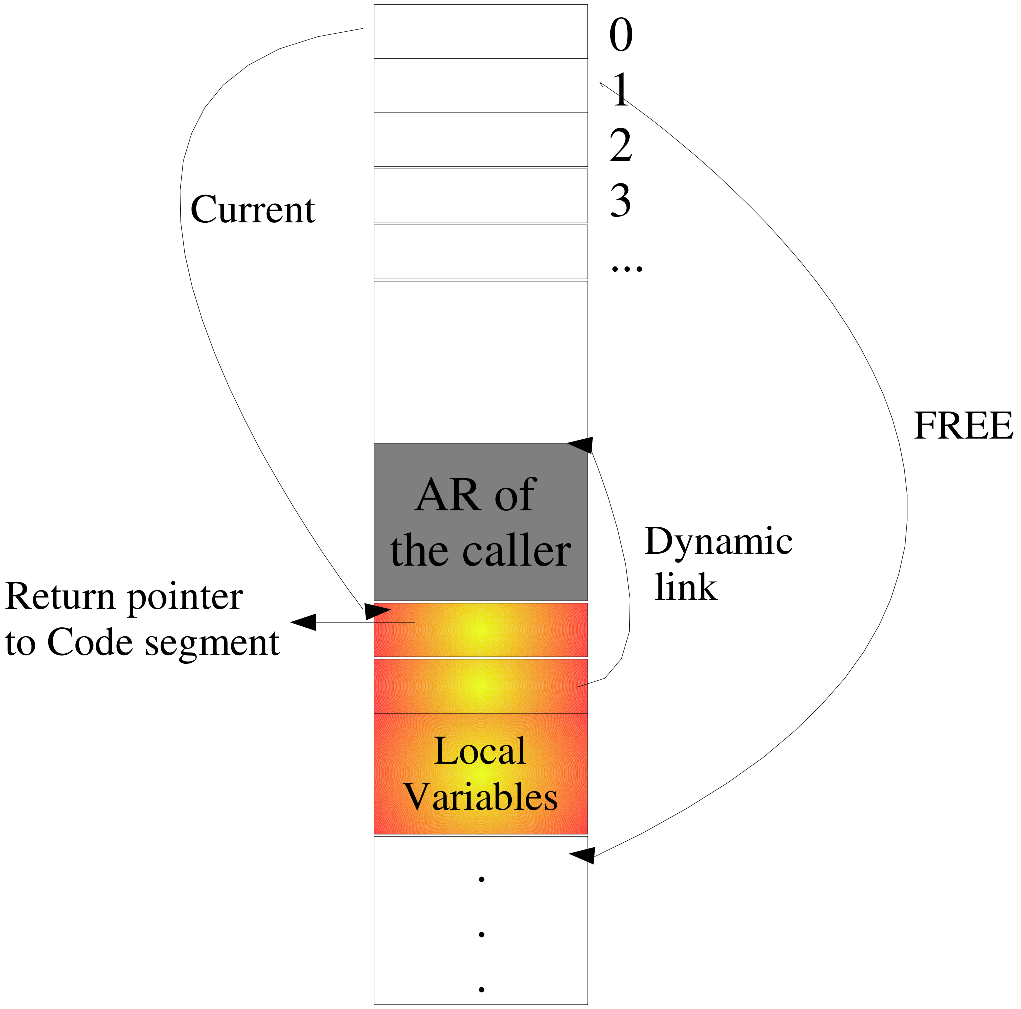
Activation Records

- The cell at offset 0 of the activation record is the *return pointer*. The return pointers points to the instruction to be executed upon return from a function.
- The cell at offset 1 is the *dynamic link*. Dynamic link points to the base address of the caller's activation record. The chain of the dynamic links is called *dynamic chain*

C3 return value

- We need space to store the returned value
- The routine’s activation record is deallocated upon termination, so the return value cannot be stored in it
- Therefore, the caller’s activation record is extended to provide the required space
- The callee uses a negative offset to write the returned value

C3


0
1
2
3
...
Current
FREE
AR of the caller
Dynamic link
Return pointer to Code segment
Local Variables
.
.
.

Calling sequence

	Step	Meaning
1	set 1, D[1] + 1	Allocate space for the return value
2	set D[1], ip + 4	Set the value of the return pointer in the callee's activation record
3	set D[1]+1, D[0]	Set the dynamic link
4	set 0, D[1]	Set CURRENT
5	set 1,D[1] + size(AR)	Set FREE
6	jump start_address	Jump to the collee's code segment

Return sequence

	Step	Meaning
1	set D[0]-1, ret. value	Set return value
2	set 1, D[0] – 1	Set FREE
3	set 0, D[D[0] + 1]	Set CURRENT
4	jump D[D[1]] + 1	Jump to the stored return pointer

Reference

- Ghezzi, C., and Jazayeri M. *Programming Language Concepts*. 3rd ed. New York: John Wiley and Sons. 1998.