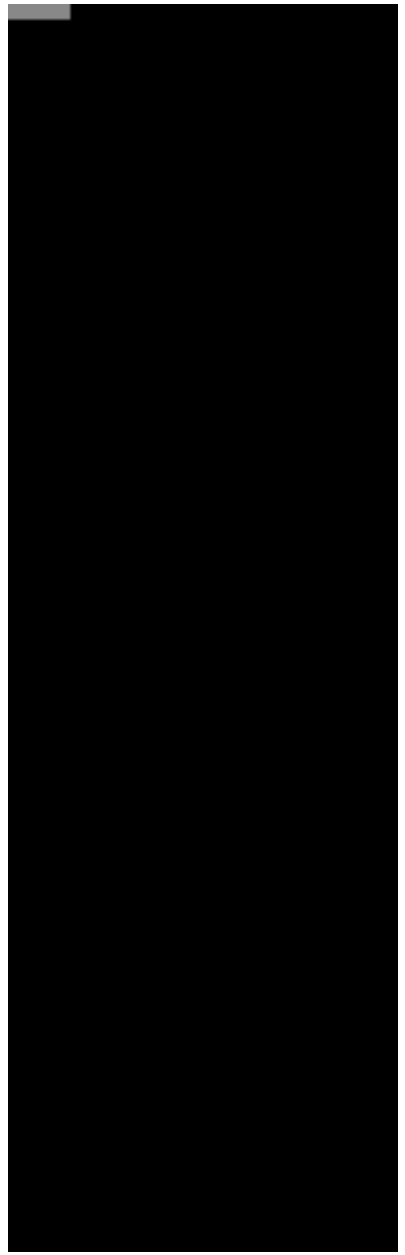
Reference counting; mark & sweep

Simplest approach: no GC

Wait until the end of the task / program

Deallocate everything in one go

Used by, eg, Apache web server



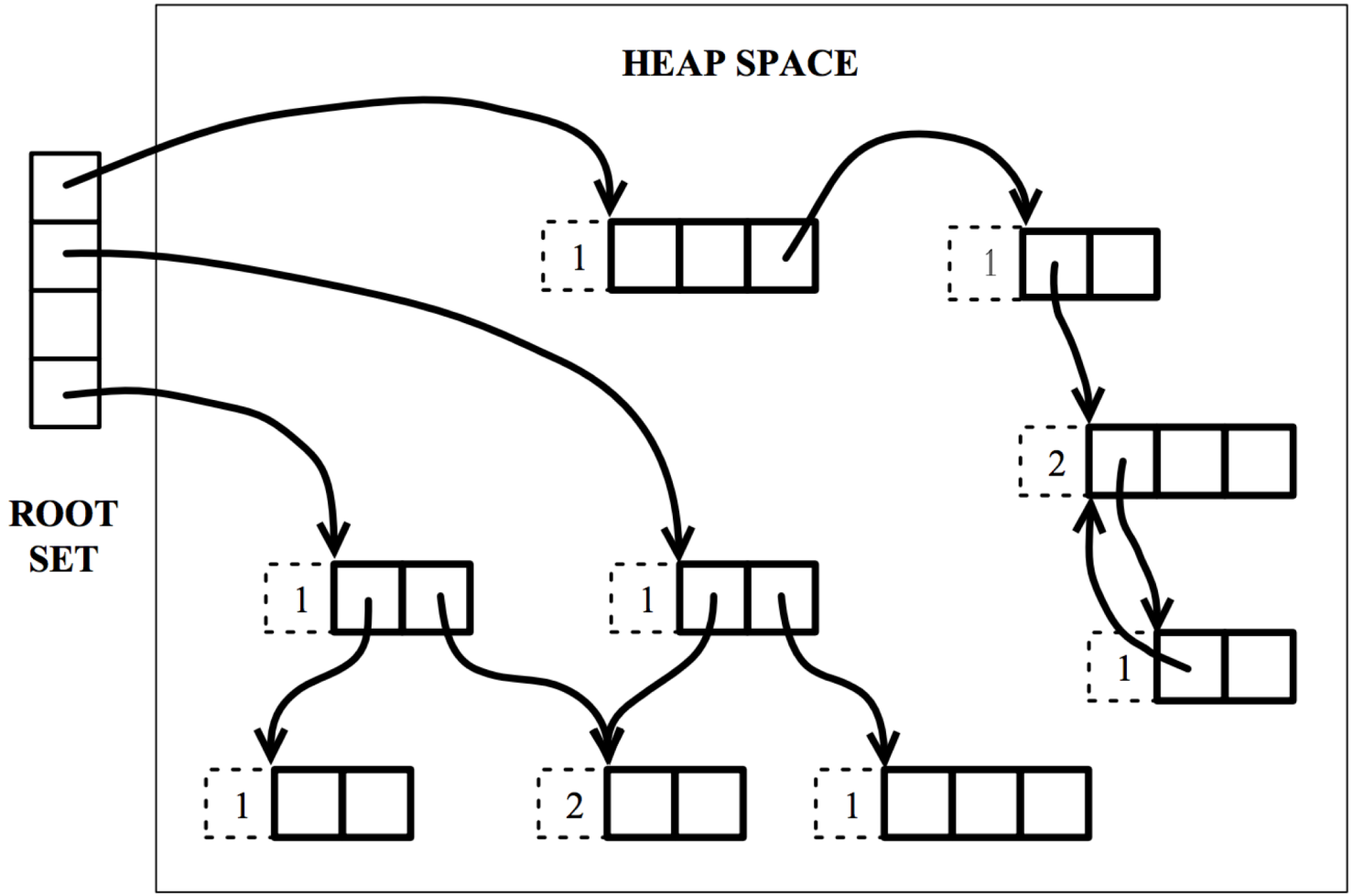
Distinguishing live objects from garbage

Reference counting: each object has an associated count of the references to it

Marking: recursively trace from the root set and mark live objects

Copying: move all live objects into another part of the heap. Anything not moved is known to be garbage

Reference counting

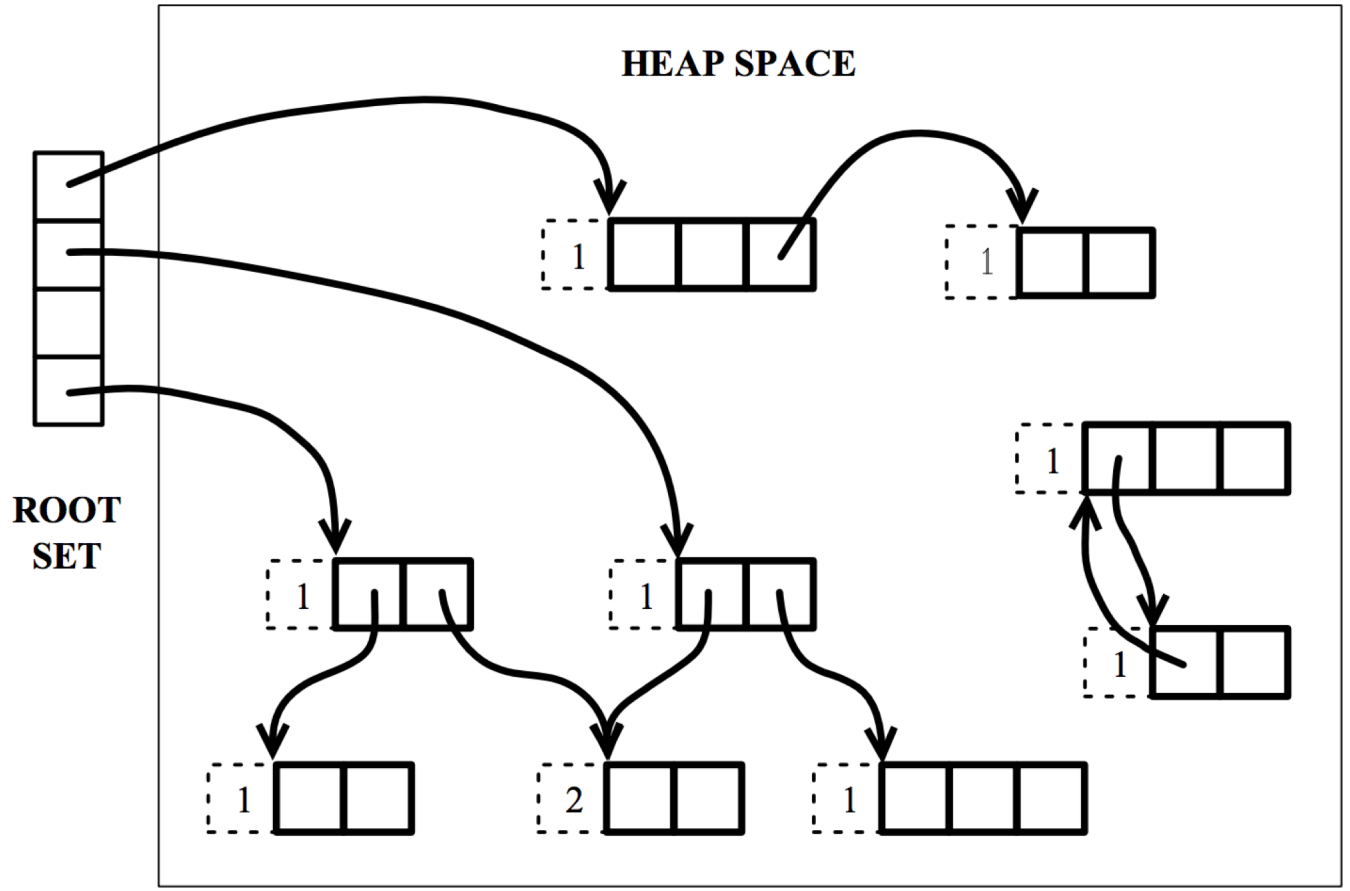


Is it ever possible to do automated code analysis?

```
(define v (make-vector 1000))
(define k (vector-length v))
;; rest of program
```

```
(local ([define v (make-vector 1000)]
        [define k (vector-length v)])
;; some code
)
```

Problems with reference counting



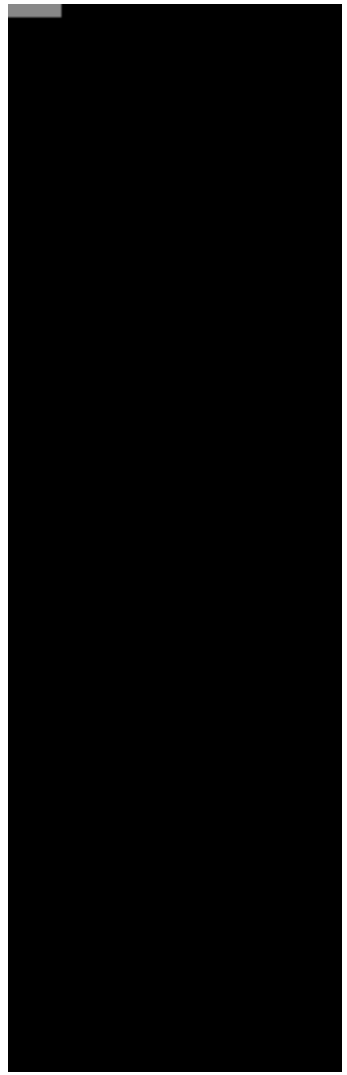
Alternative: deferred reference counting

- Only keep count references from one heap object to another
- Don’t count references from variables (especially stack)
  - Counts may be wrong, so don’t delete immediately when down to 0
- Periodically scan the stack to update counts, then clean up

- Avoids overhead for lots of local pointer assignments
- Sweep often enough to clean up
  - but infrequently enough to be efficient

Mark and sweep

- Distinguish live objects from garbage
  - Reset all marks
  - Start at the root set
  - Traverse the tree recursively, marking all reachable objects
- Reclaim the garbage
  - Sweep through the heap
  - Reclaim any unmarked objects
  - Reclaimed objects are put on “free lists”



Problems with Mark and Sweep

- Fragmentation
- Cost is proportional to size of the heap (number of objects)
- Locality of reference
  - totally thrashes cache, virtual memory, etc.

Mark and compact

- Start by marking live objects
- Live objects are then compacted
  - Just slide them down!
  - No full sweep through heap
- A type of “copying collector”

- New objects can be allocated efficiently

- Still requires several passes through the data
  - One pass to compute new locations
  - One pass to update references
  - One pass to actually move the data