

```

int link[n], end[n];
process Find[i = 0 to n-1] {
  int new, d = 1;
  end[i] = link[i]; /* initialize elements of end */
  barrier(i);
  ## FIND: end[i] == index of end of the list
  ##          at most  $2^{d-1}$  links away from node i
  while (d < n) {
    new = null; /* see if end[i] should be updated */
    if (end[i] != null and end[end[i]] != null)
      new = end[end[i]];
    barrier(i);
    if (new != null) /* update end[i] */
      end[i] = new;
    barrier(i);
    d = d + d; /* double the distance */
  }
}

```

**Figure 3.18** Finding the end of a serially linked list.