

(b)

```
1  iter CyclicArr.these(param tag: iterKind, followThis, param fast: bool = false) var
2      where tag == iterKind.follower {
3
4      //check that all elements in chunk are from the same locale by examining each dim
5      for i in 1..rank {
6          if (followThis(i).stride * dom.whole.dim(i).stride %
7              dom.dist.targetLocDom.dim(i).size != 0) {
8              //call original follower iterator helper for nonlocal elements
9              for i in followThis {
10                 yield accessHelper(i);
11             }
12         }
13         if arrSection.locale.id == here.id then local {
14             //call original fast follower iterator helper for local elements
15         } else {
16             //allocate local buffer to hold remote elements, compute source and destination
17             //strides, number of elements to communicate
18             chpl_comm_gets(buf, deststr, arrSection.myElems._value.theData, srcstr, count);
19             var changed = false;
20             for i in buf {
21                 var old_i = i;
22                 yield i;
23                 var new_val = i;
24                 if(old_val != new_val) then changed = true;
25             }
26             if changed then
27                 chpl_comm_puts(arrSection.myElems._value.theData, srcstr, buf, deststr, count);
28         }
29     }
```