

```

1 void cilkplus_an_mandel(
2     int n,
3     std::complex c[n],
4     int count[n],
5     int max_count
6 ) {
7     std::complex z[n];
8     int test[n];
9     z[:] = 0;
10    for (int k = 0; k < max_count; k++) {
11        // test for divergence for all pixels in chunk
12        test[:] = (abs(z[:]) < 2.0);
13        if (0 == __sec_reduce_add(test[:])) {
14            // terminates loop only if all have diverged
15            break;
16        }
17        // increment counts only for pixels that have not diverged
18        count[:] += test[:];
19        // unconditionally update state of iteration
20        z[:] = z[:] * z[:] + c[:];
21    }
22 }
23
24 void cilkplus_mandel(
25     int p[][],
26     int max_row,
27     int max_col,
28     int depth
29 ) {
30     // parallelize over rows
31     cilk_for (int i = 0; i < max_row; ++i)
32         // loop over the row in chunks of 8
33         for (int j = 0; j < max_col; j += 8)
34             // compute the Mandelbrot counts for a chunk
35             cilkplus_an_mandel(8, p[i]+j, points[i]+j, depth);
36 }

```

LISTING 4.14

Mandelbrot in Cilk Plus using `cilk_for` and array notation for explicit *vectorization*.