

## CSci 280, 27 Aug 2014

```
1 public static int isqrt1(int n) {
2     int cur = 0;
3     while (cur * cur <= n) {
4         cur++;
5     }
6     return cur;
7 }
8
9 public static int isqrt2(int n) {
10    int lo = 0;
11    int hi = n + 1;
12    while (hi - lo > 1) {
13        int mid = (lo + hi) / 2;
14        if (mid * mid <= n) lo = mid;
15        else hi = mid;
16    }
17    return lo;
18 }
19
20 public static boolean areAllUnique(int[] nums) { // n = nums.length
21     for (int i = 0; i < nums.length; i++) {
22         for (int j = i + 1; j < nums.length; j++) {
23             if (nums[i] == nums[j]) return false;
24         }
25     }
26     return true;
27 }
28
29 public static void cutLarge(double[] scores) { // n = scores.length
30     double total = 0.0;
31     for (int i = 0; i < scores.length; i++) {
32         total += scores[i];
33     }
34     double avg = total / scores.length;
35     for (int j = 0; j < scores.length; j++) {
36         if (scores[j] > avg) scores[j] = avg;
37     }
38 }
39
40 public static int mystery(int n) {
41     int i = 0;
42     int j = 0;
43     while (j <= n) {
44         i++;
45         j += i + i - 1;
46     }
47     return i - 1;
48 }
```

```

50 public static int factor(int n) {
51     int i = 0;
52     int j = 0;
53     int k = 0;
54     while (j < n && k != n) {
55         if (i < j) {
56             i++;
57             k += j;
58         } else {
59             i = 0;
60             j++;
61             k = 0;
62         }
63     }
64     return j;
65 }
66
67 public static int countLatticePoints(int r) {
68     int count = 0;
69     int y = 0;
70     for (int x = r; x > 0; x--) {
71         while (x*x + y*y < r*r) {
72             y++;
73         }
74         count += y;
75     }
76     return 4 * count + 1;
77 }
78
79 public static int countPrimesTo(int n) {
80     int count = 0;
81     boolean[] ok = new boolean[n + 1];
82     Arrays.fill(ok, true);
83     for (int i = 2; i <= n; i++) {
84         if (ok[i]) {
85             count++;
86             for (int j = i + i; j <= n; j += i) {
87                 ok[j] = false;
88             }
89         }
90     }
91     return count;
92 }

```