/home/gandrade/Manual-da-Sarrada/drazy/regular_chulltrick.cpp
Page 1 of 1 Qui 10 Nov 2016 15:13:26 BRST

```
typedef long double ld;
 1
2
3
     struct node {
    ld left, p, q;
 4
5
     vector<node> lines;
 6
7
     ld get inter(ld p, ld q, ld r, ld s) { return (q-s)/(r-p); }
 8
 9
     // lines must be inserted in strictly decreasing order of p
     void insert(ld p, ld q) {
    while (sz(lines) >= 2) {
        ld r = lines[sz(lines)-2].p;
}
10
11
12
               ld s = lines[sz(lines)-2].q;
13
               if (lines.back().left > get inter(p,q,r,s)) lines.pop back();
14
15
               else break;
16
17
          node nd;
18
          nd.p = p; nd.q = q;
          nd.left = (lines.empty() ? -INFINITY : get inter(lines.back().p,
19
          lines.back().q, p, q));
20
          lines.pb(nd);
21
     }
22
23
     ld get min(ld x) {
24
25
          node aux;
          aux.left = x;
26
          auto nd = *prev(upper bound(all(lines), aux, [](node a, node b){ return a.left →
          < b.left; }));
27
          return nd.p*x + nd.q;
28
     }
29
```