

李超線段樹

李超線段樹

題目 (動態凸包)

現在有 Q 個操作，每個操作會是以下兩種中的一種：

- 加入一條直線 $y = mx + k$
- 詢問在 $x = t$ 處最大的 y 值
- $1 \leq Q \leq 10^5$
- $|m|, |k| \leq 10^9$
- $1 \leq t \leq 10^5$

李超線段樹

用 set 維護上凸包上的線段，維護線段控制的左右界，每次加入直線先搜他控制的區間，往左右殺掉其他線段，查詢的時候二分搜是哪條線段代值進去。注意 iterator 使用、全整求線交點.....

太麻煩了，而且常數不小，而且我沒寫過
有沒有簡單一點的辦法？

李超線段樹

李超線段樹：

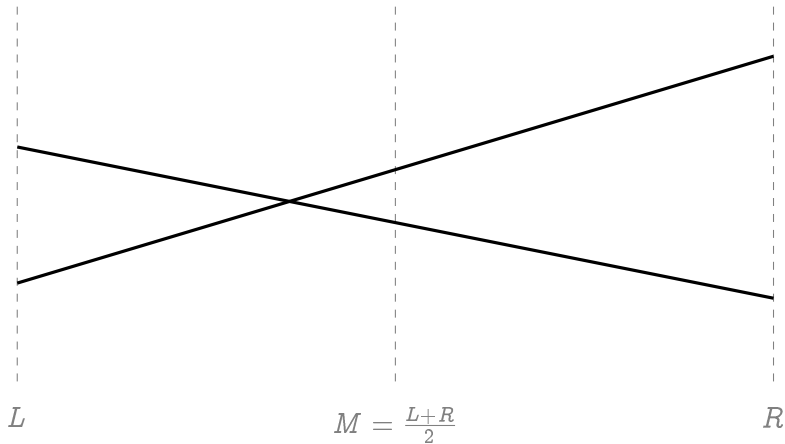
- 對要查詢的**值域**開線段樹，葉子代表單一個 x 的值
- 每個節點存**一條**對**中點**來說 y 最大的直線
 - 對中點一定是有用的
 - 可能還對這個區間的其他一部分有用

李超線段樹 – 插入直線

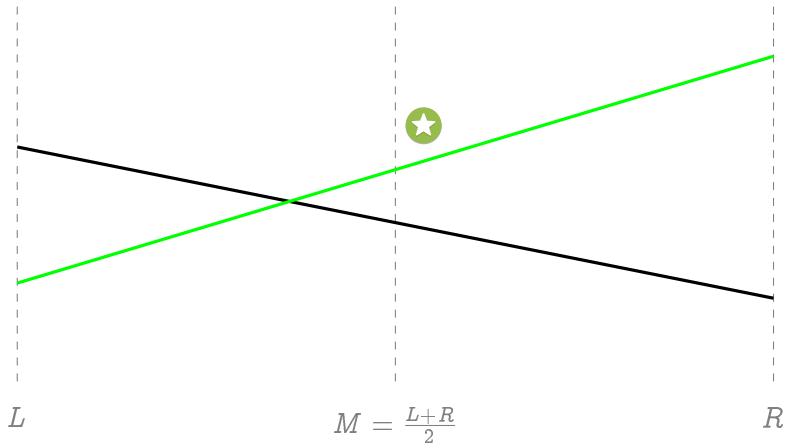
原本節點上有一條直線
這次詢問想插入另外一條直線

一個節點只能存一條線誰要留下來？另一條線要去哪裡？

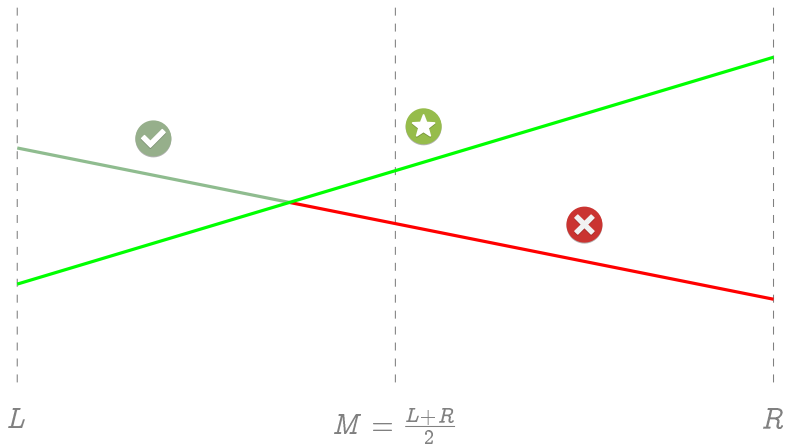
李超線段樹 – 插入直線



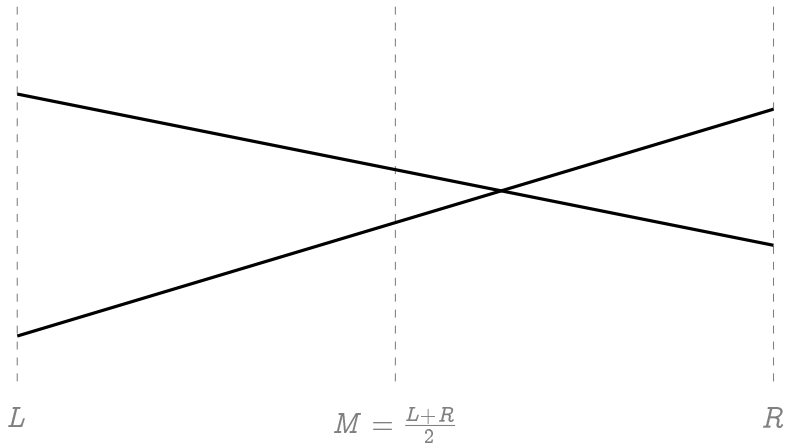
李超線段樹 – 插入直線



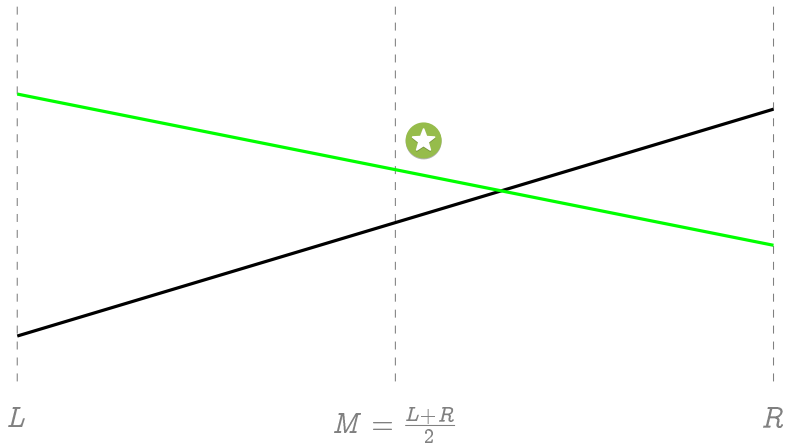
李超線段樹 – 插入直線



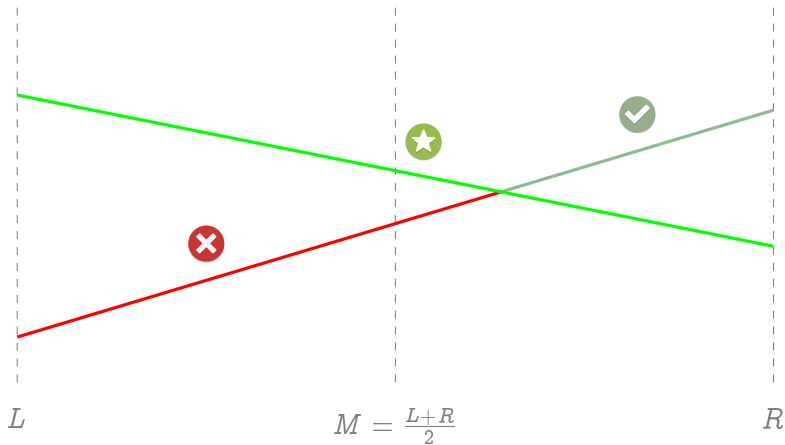
李超線段樹 – 插入直線



李超線段樹 – 插入直線



李超線段樹 – 插入直線



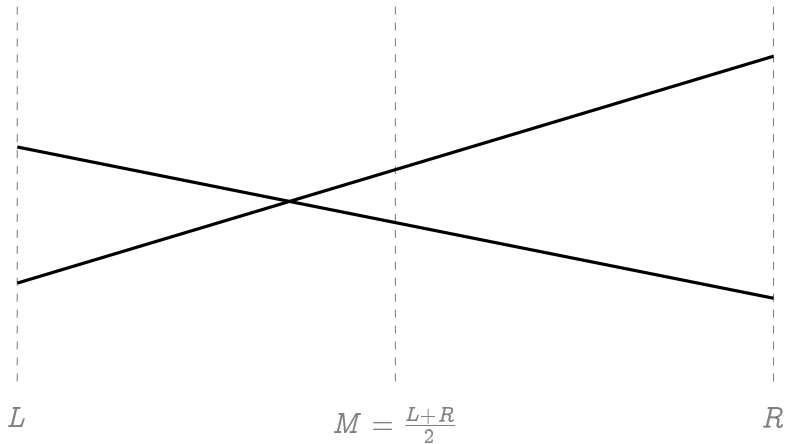
李超線段樹 – 插入直線

一條直線在中點輸掉之後不能直接扔掉，因為他還沒輸光，區間內某些 x 的範圍可能還需要他

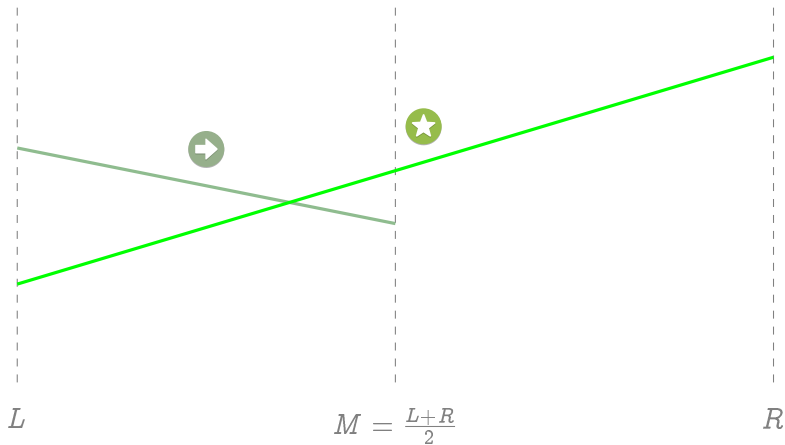
在中點輸掉的話，一定也會在左右其中一邊輸光
只有其中一邊可能還會需要用到這條直線，遞迴把他交給線段樹上那半邊的子樹處置，另外半邊已經不需要他了

到葉子還輸的話那這條線徹底不會被任何人需要

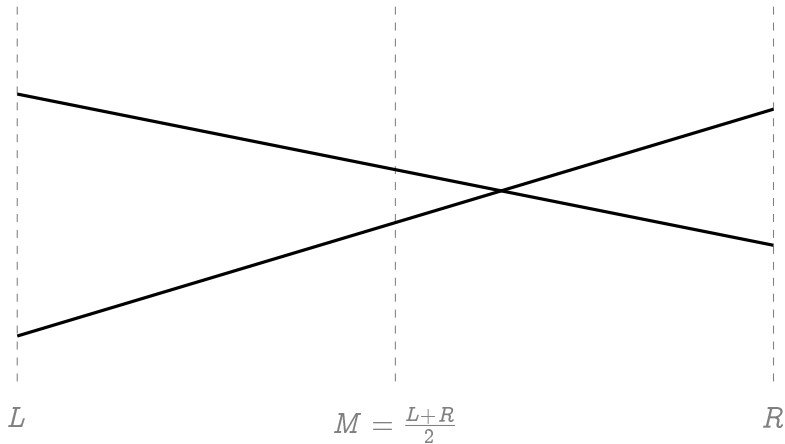
李超線段樹 – 插入直線



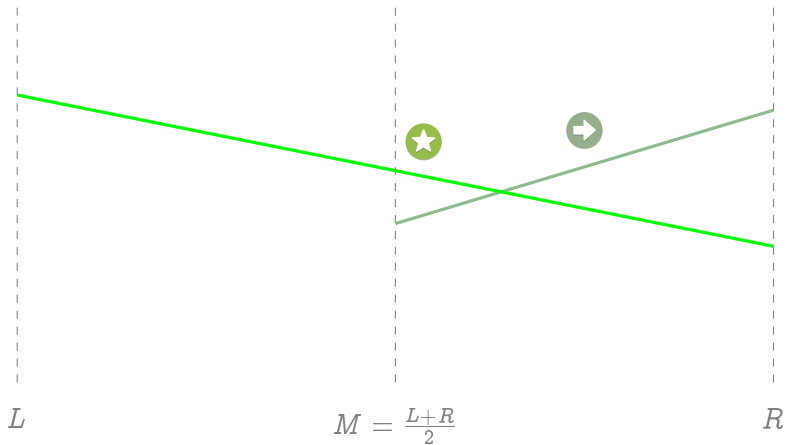
李超線段樹 – 插入直線



李超線段樹 – 插入直線



李超線段樹 – 插入直線



李超線段樹 – 插入直線

從根節點出發，到葉節點為止：

- 代中點 x 座標比較兩條直線，贏家留在節點上
- 比較兩條直線的斜率
 - 如果贏家的斜率比較**大**，輸家往**左**子樹遞迴插入
 - 如果贏家的斜率比較**小**，輸家往**右**子樹遞迴插入

一直往子樹丟包直線

時間複雜度 $O(\text{線段樹高}) = O(\log N)$

李超線段樹 – 單點查詢

直線被扔到隔壁節點，代表這個範圍的 x 全都用不到這條直線
一個 x 可能用到的直線，都存在他的祖先們身上

- 找到代表這個 x 值的葉子
- 檢查所有祖先存的直線，每個都代一次，回答最大的 y

時間複雜度 $O(\text{線段樹高}) = O(\log N)$

李超線段樹 – 實做

包裝直線作為函數使用

```
1 struct Line {  
2     int a, b; //  $y = ax + b$   
3     Line(int _a = 0, int _b = 0): a(_a), b(_b) {}  
4     int operator()(int x) { return a * x + b; }  
5 };
```

李超線段樹 – 實做

插入直線

- 代中點 x 座標比較兩條直線，贏家留在節點上
- 比較兩條直線的斜率
- 遞迴插入

```
1 void insert(int id, int l, int r, Line ln) {  
2     int m = (l + r) / 2;  
3     if(lns[id](m) < ln(m)) swap(lns[id], ln);  
4     if(l == r) return;  
5     if(lns[id].a > ln.a) insert(L(id), l, m, ln);  
6     else insert(R(id), m + 1, r, ln);  
7 }
```

李超線段樹 – 實做

單點查詢

- 找到代表這個 x 值的葉子
- 檢查所有祖先存的直線，每個都代一次，回答最大的 y

```
1  int qry(int id, int l, int r, int x) {
2      int m = (l + r) / 2;
3      int res = lns[id](x);
4      if(l == r) return res;
5      if(x <= m) res = max(res, qry(L(id), l, m, x));
6      else res = max(res, qry(R(id), m + 1, r, x));
7      return res;
8  }
```

李超線段樹 – 座標壓縮

題目 (Line Add Get Min , Library Checker)

你有 N 條直線 $y = a_i x + b_i$ 。請你處理 Q 個詢問：

- 加入一條直線 $y = ax + b$
- 詢問 $x = p$ 處最小的 y 值
- $1 \leq N, Q \leq 2 \times 10^5$
- $|a_i|, |p| \leq 10^9$
- $|b_i| \leq 10^{18}$

李超線段樹 – 座標壓縮

題目 (Line Add Get Min , Library Checker)

你有 N 條直線 $y = a_i x + b_i$ 。請你處理 Q 個詢問：

- 加入一條直線 $y = ax + b$
- 詢問 $x = p$ 處最小的 y 值
- $1 \leq N, Q \leq 2 \times 10^5$
- $|a_i|, |p| \leq 10^9$ **???!**
- $|b_i| \leq 10^{18}$

李超線段樹 – 座標壓縮

剛剛對 x 的值域 $1, 2, \dots, 10^5$ 開線段樹

現在事先收集會被詢問到的 x 座標
對會被問到的 x 開線段樹

詢問是浮點數的時候也可以

李超線段樹 – 座標壓縮

如果事先不知道詢問位置呢？

李超線段樹 – 座標壓縮

如果事先不知道詢問位置呢？

動態開點，用不到的節點不要理他

李超線段樹 – 插入線段

如果插入的不是直線，而是有左右範圍限制的線段呢？

$q(\text{'}\square\text{'})$ 插圖

李超線段樹 – 插入線段

一般線段樹是怎麼做區間修改的？

李超線段樹 – 插入線段

一般線段樹是怎麼做區間修改的？

找 $O(\log N)$ 個節點覆蓋詢問的區間，修改那些節點

李超線段樹 – 插入線段

一般線段樹是怎麼做區間修改的？

找 $O(\log N)$ 個節點覆蓋詢問的區間，修改那些節點

找 $O(\log N)$ 個節點覆蓋線段範圍，對那些節點插入直線

李超線段樹 – 插入線段


一般線段樹是怎麼做區間修改的？

找 $O(\log N)$ 個節點覆蓋詢問的區間，修改那些節點

找 $O(\log N)$ 個節點覆蓋線段範圍，對那些節點插入直線

時間複雜度：插入一次 $O(\log N)$ ，總共 $O(\log^2 N)$

李超線段樹 – 應用

- 斜率優化 

李超線段樹 – 應用

- 斜率優化 ✓
- 四邊形優化 ✓

不只是直線，有**優超性**的函數都可以