note

 \blacktriangleright

49 views

July 16 BinarySearchModels

```
1 // 非严格递增
2 // 模板2 f(a[pos]) < target <= f(a[pos + 1])
 1 // 非严格递增
 2 // 模板1 f(a[pos]) <= target < f(a[pos + 1])
 3 * public int func(input...) {
                                                        3 v public int func(input...) {
       // 经过分析,将原问题转为
                                                              // 经过分析,将原问题转为
                                                        4
        // 白板上的东西
                                                               // 白板上的东西
        T[] arr = new T[n];
                                                               T[] arr = new T[n];
 6 *
                                                        6 ₹
                                                               G target = xxx;
        G target = xxx;
        int start = 0;
                                                               int start = 0;
 8
                                                        8
        int end = n - 1;
int pos = -1;
                                                               int end = n - 1;
 9
                                                       9
                                                               int pos = -1;
10
                                                       10
        while (start <= end) {
                                                               while (start <= end) {
11 ▼
                                                       11 v
12
           int mid = start + (end - start) / 2;
                                                      12
                                                                int mid = start + (end - start) / 2;
           G midVal = g(mid);
if (midVal <= target) {
                                                                  G midVal = g(mid);
if (midVal < target) {</pre>
13
                                                       13
14 +
                                                       14 ▼
                                                                   pos = mid;
15
              pos = mid;
                                                       15
                start = mid + 1;
                                                       16
                                                                       start = mid + 1;
16
            } else {
                                                       17 🔻
                                                                   } else {
17 ▼
               end = mid - 1;
                                                       18
                                                                      end = mid - 1;
18
                                                       19
19
                                                       20
20
21
        return pos;
                                                       21
                                                               return pos;
                                                       22 }
22 }
 1 // 非严格递增
2 // 模板3 f(a[pos - 1]) <= target < f(a[pos])
                                                       3 * public int func(input...) {
                                                               // 经过分析,将原问题转为
                                                        4
                                                                // 白板上的东西
        T[] arr = new T[n];
G target = xxx;
 6 ₹
                                                        6 ▼
                                                               T[] arr = new T[n];
                                                               G target = xxx;
                                                        7
 8
        int start = 0;
                                                        8
                                                               int start = 0;
 9
        int end = n - 1;
                                                        Q
                                                               int end = n - 1;
                                                               int pos = n;
10
        int pos = n;
                                                       10
        while (start <= end) {
                                                               while (start <= end) {
11 v
                                                       11 +
            int mid = start + (end - start) / 2;
                                                       12
                                                                int mid = start + (end - start) / 2;
                                                                   G midVal = g(mid);
            G midVal = g(mid);
                                                       13
            if (midVal > target) {
                                                                   if (midVal >= target) {
                                                       14 ▼
                                                                    pos = mid;
             pos = mid;
                                                       15
                end = mid - 1;
                                                       16
                                                                       end = mid - 1;
            } else {
                                                       17 🔻
                                                                   } else {
               start = mid + 1;
                                                       18
                                                                      start = mid + 1;
                                                       19
19
                                                       20
20
21
        return pos;
                                                       21
                                                               return pos;
                                                       22 }
22 }
                                                        1 // 非严格递减
2 // 模板6 f(a[pos]) > target >= f(a[pos + 1])
3 r public int func(input...) {
 1 // 非严格递减
 2 // 模板5 f(a[pos]) >= target > f(a[pos + 1])
 3 v public int func(input...) {
       // 经过分析,将原问题转为
// 白板上的东西
                                                               // 经过分析,将原问题转为
// 白板上的东西
                                                          4
        T[] arr = new T[n];
                                                                 T[] arr = new T[n];
        G target = xxx;
                                                                 G target = xxx;
        int start = 0;
                                                                 int start = 0;
        int end = n - 1;
                                                                 int end = n - 1;
        int pos = -1;
                                                                 int pos = -1;
10
                                                         10
        while (start <= end) {
                                                                 while (start <= end) {
11 •
                                                         11 +
           int mid = start + (end - start) / 2;
G midVal = g(mid);
if (midVal >= target) {
                                                                    int mid = start + (end - start) / 2;
12
                                                         12
                                                                    G midVal = g(mid);
if (midVal > target) {
13
                                                         13
                                                         14 +
14 +
              pos = mid;
                                                                      pos = mid;
                                                         15
15
                start = mid + 1;
                                                                         start = mid + 1;
16
                                                         16
                                                         17 -
17 •
            } else {
                                                                     } else {
               end = mid - 1:
                                                                        end = mid - 1:
                                                         18
18
                                                                     }
19
                                                         19
20
                                                         20
21
        return pos;
                                                        21
                                                                 return pos;
                                                        22 }
```

```
1 // 非严格递减
2 // 模板7
                                                         1 // 非严格递减
                                                         2 // 模板8
                 f(a[pos - 1]) >= target > f(a[pos])
                                                                          f(a[pos - 1]) > target >= f(a[pos])
                                                         3 * public int func(input...) {
       // 经过分析,将原问题转为
// 白板上的东西
 6 ₹
        T[] arr = new T[n];
                                                         6 ₹
                                                                 T[] arr = new T[n];
        G target = xxx;
                                                                 G target = xxx;
 8
        int start = 0;
                                                         8
                                                                 int start = 0;
 9
        int end = n - 1;
                                                         9
                                                                 int end = n - 1;
10
        int pos = n;
                                                         10
                                                                 int pos = n;
11 •
        while (start <= end) {
                                                         11 •
                                                                 while (start <= end) {
            int mid = start + (end - start) / 2;
                                                                     int mid = start + (end - start) / 2;
12
                                                         12
            G midVal = g(mid);
if (midVal < target) {</pre>
                                                                    G midVal = g(mid);
if (midVal <= target) {</pre>
13
                                                         13
14 🔻
                                                         14 🔻
15
               pos = mid;
                                                         15
                                                                        pos = mid;
                end = mid - 1;
                                                                         end = mid - 1;
16
                                                         16
17 🔻
            } else {
                                                         17 🔻
                                                                     } else {
                start = mid + 1;
                                                                        start = mid + 1;
18
                                                         18
                                                                     }
19
                                                        19
20
                                                         20
21
                                                         21
        return pos;
                                                                 return pos;
22 }
                                                        22 }
```

八个模板如上,递增情况下,当midVal > target时,end = mid - 1;,砍掉后头到前面值更小的区间内找target,当midVal < target时,砍掉前头到后面值更大的区间内找target。递减情况下,当midVal > target时,砍掉前头到后面值更大的区间找target。当midVal < target时,砍掉后头到前面值更大的区间找target。pos的更新永远跟着if(condition)后面走。

如有错误,欢迎指正。谢谢大家。

binary_search

Updated 8 days ago by Chenhui Li and New Soft Valley

followup discussions for lingering questions and comments



New Soft Valley 9 days ago

Resolved Unresolved

2, 3, 6, 7不对

if (condition) {
 pos = mid;
 调整start或者end
} else {
 调整start或者end

pos = mid永远发生在if里面。不会在else里面。

Chenhui Li 9 days ago 终于理解了,还是昨晚说的那个问题,把要求什么搞混了,要求的是什么,if(condition)就写什么,pos = mid不变,然后start end根据欲前面还是后面来调整。



New Soft Valley 9 days ago

那快更新啊!这么多双眼睛盯着你们的代码!



New Soft Valley 9 days ago 第5,6的pos,应该是-1

另外每个模板里,把T[] arr = new T[size]; 改写成T[] arr = new T[n];
把int end = arr.length - 1; 改成int end = n - 1;

会方便以后杳看一些



Bre 8 days ago 所有模版都调对吗? 好像chenhui 还改了。



Chenhui Li 8 days ago 不好意思,刚刚改完了。



New Soft Valley 8 days ago

Ship-it!