

트리 2 (연습)

소스코드

최백준 [choi@startlink.io](mailto:choi@startlink.io)

---

C++14

```
1 #include <iostream>
2 using namespace std;
3 int d[200001][20];
4 int main() {
5     ios_base::sync_with_stdio(false);
6     cin.tie(nullptr);
7     int n;
8     cin >> n;
9     for (int i=1; i<=n; i++) {
10         cin >> d[i][0];
11     }
12     for (int j=1; j<20; j++) {
13         for (int i=1; i<=n; i++) {
14             d[i][j] = d[d[i][j-1]][j-1];
15         }
16     }
17     int q;
18     cin >> q;
19     while (q--) {
20         int m, x;
21         cin >> m >> x;
22         /* f_m(x)
23         for (int i=0; i<20; i++) {
24             if (m & (1 << i)) {
25                 x = d[x][i];
26             }
27         }
28         cout << x << '\n';
29     }
30     return 0;
31 }
32
```

*Handwritten notes:*

- Under line 16:  $\text{LCA}$
- Under line 22:  $m = 101101$  with arrows pointing to bits 0, 1, 2, 3, 4, 5. Below it:  $2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 = m$
- Under line 25:  $f_{2^i}(x)$  in a box, with an arrow pointing from the assignment `x = d[x][i];`

결과	메모리	시간	코드 길이
맞았습니다!!	17612 KB	240 ms	613 B

C++14

```
1 #include <cstdio>
2 #include <algorithm>
3 #include <iostream>
4 #include <vector>
5 #include <set>
6 #include <map>
7 #include <queue>
8 #include <stack>
9 #include <sstream>
10 using namespace std;
11 const int MAX = 100111;
12 vector<pair<int,int>> a[MAX];
13 int parent[MAX];
14 bool check[MAX];
15 int depth[MAX];
16 int p[MAX][18];
17 int len[MAX];
18 int len_min[MAX][18];
19 int len_max[MAX][18];
20 pair<int,int> lca(int u, int v) {
21     if (depth[u] < depth[v]) {
22         swap(u,v);
23     }
24     int ans_min = 10000000;
25     int ans_max = 0;
26     int log = 1;
27     for (log=1; (1<=log) <= depth[u]; log++);
28     log--;
29     for (int i=log; i>=0; i--) {
30         if (depth[u] - (1<=i) >= depth[v]) {
31             if (ans_min > len_min[u][i]) {
32                 ans_min = len_min[u][i];
33             }
34             if (ans_max < len_max[u][i]) {
35                 ans_max = len_max[u][i];
36             }
37             u = p[u][i];
38         }
39     }
40     //printf("! %d %d\n",u,v);
41     //printf("@ %d %d\n",ans_min,ans_max);
42     if (u == v) {
43         return make_pair(ans_min,ans_max);
44     } else {
45         for (int i=log; i>=0; i--) {
46             if (p[u][i] != 0 && p[v][i] != p[v][i]) {
47                 if (ans_min > len_min[u][i]) {
48                     ans_min = len_min[u][i];
49                 }
50                 if (ans_max < len_max[u][i]) {
51                     ans_max = len_max[u][i];
52                 }
53                 if (ans_min > len_min[v][i]) {
54                     ans_min = len_min[v][i];
55                 }
56                 if (ans_max < len_max[v][i]) {
57                     ans_max = len_max[v][i];
58                 }
59                 u = p[u][i];
60                 v = p[v][i];
61             }
62         }
63         if (ans_min > len_min[u][0]) {
64             ans_min = len_min[u][0];
65         }
66         if (ans_max < len_max[u][0]) {
67             ans_max = len_max[u][0];
68         }
69         if (ans_min > len_min[v][0]) {
70             ans_min = len_min[v][0];
71         }
72         if (ans_max < len_max[v][0]) {
73             ans_max = len_max[v][0];
74         }
75         return make_pair(ans_min,ans_max);
76     }
77 }
78 int main() {
79     int n;
80     scanf("%d",&n);
81     for (int i=0; i<n-1; i++) {
82         int u,v,w;
83         scanf("%d %d %d",&u,&v,&w);
84         a[u].push_back(make_pair(v,w));
85         a[v].push_back(make_pair(u,w));
86     }
87     depth[1] = 0;
88     check[1] = true;
89     queue<int> q;
90     q.push(1);
91     parent[1] = 0;
92     while (!q.empty()) {
93         int x = q.front();
94         q.pop();
95         for (auto p : a[x]) {
96             int y = p.first;
97             if (!check[y]) {
98                 depth[y] = depth[x] + 1;
99                 check[y] = true;
100                 parent[y] = x;
101                 len[y] = p.second;
102                 q.push(y);
103             }
104         }
105     }
106     for (int i=1; i<=n; i++) {
107         p[i][0] = parent[i];
108         len_min[i][0] = len[i];
109         len_max[i][0] = len[i];
110     }
111     for (int j=1; (1<=j) < n; j++) {
112         for (int i=1; i<=n; i++) {
113             if (p[i][j-1] != 0) {
114                 p[i][j] = p[p[i][j-1]][j-1];
115                 len_min[i][j] = len_min[i][j-1];
116                 len_max[i][j] = len_max[i][j-1];
117                 if (p[p[i][j-1]][j-1] != 0) {
118                     len_min[i][j] = min(len_min[i][j-1], len_min[p[i][j-1]][j-1]);
119                     len_max[i][j] = max(len_max[i][j-1], len_max[p[i][j-1]][j-1]);
120                 }
121             }
122             //printf("%d %d %d %d\n",i,j,len_min[i][j],len_max[i][j]);
123         }
124     }
125 }
126 int m;
127 scanf("%d",&m);
128 while (m--) {
129     int u, v;
130     scanf("%d %d",&u,&v);
131     auto p = lca(u,v);
132     printf("%d %d\n",p.first,p.second);
133 }
134 return 0;
135 }
```

결과

메모리

시간

코드 길이

맞았습니다!!

30416 KB

220 ms

3608 B

끝

---

# 코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 [codeplus@startlink.io](mailto:codeplus@startlink.io) 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.