

이름 검색

소스코드

최백준 choi@startlink.io



C++14

```
1 #include <iostream>
2 using namespace std;
3 long long calc(int n) {
4     long long ans = 0;
5     for (int start=1, len=1; start<=n; start*=10, len++) {
6         int end = start*10-1;
7         if (end > n) {
8             end = n;
9         }
10        ans += (long long)(end - start + 1) * len;
11    }
12    return ans;
13 }
14 int main() {
15     int n;
16     long long k;
17     cin >> n >> k;
18     if (calc(n) < k) {
19         cout << -1 << '\n';
20         return 0;
21     }
22     int left = 1;
23     int right = n;
24     int ans = 0;
25     while (left <= right) {
26         int mid = (left+right)/2;
27         long long len = calc(mid);
28         if (len < k) {
29             left = mid+1;
30         } else {
31             ans = mid;
32             right = mid-1;
33         }
34     }
35     string s = to_string(ans);
36     long long l = calc(ans);
37     cout << s[s.length()-l-k-1] << '\n';
38     return 0;
39 }
```

Handwritten notes:

- calc function:** $S-E$ (start-end), $2/3$ (len). Diagram shows $1-9$ (len=1), $10-99$ (len=2), $100-999$ (len=3).
- main function:** $1-n$ range. 175 and 176 are shown as candidates. 176 is selected. $l-k=1$ and $l-k=2$ are noted.

결과	메모리	시간	코드 길이
----	-----	----	-------

C++14

```
1 #include <stdio>
2 long long a[10000];
3 int n, m;
4 bool check(long long x) {
5     int cnt = 0;
6     for (int i=0; i<n; i++) {
7         cnt += a[i]/x;
8     }
9     return cnt >= m;
10 }
11 int main() {
12     scanf("%d %d", &n, &m);
13     long long max = 0;
14     for (int i=0; i<n; i++) {
15         scanf("%lld", &a[i]);
16         if (max < a[i]) {
17             max = a[i];
18         }
19     }
20     long long ans = 0;
21     long long l = 1;
22     long long r = max;
23     while (l <= r) {
24         long long mid = (l+r)/2;
25         if (check(mid)) {
26             if (ans < mid) {
27                 ans = mid;
28             }
29             l = mid+1;
30         } else {
31             r = mid-1;
32         }
33     }
34     printf("%lld\n", ans);
35     return 0;
36 }
```

길이 x로 자르면
몇개의
랜선이 나오?
Yes/No

결과

메모리

시간

코드 길이

맞았습니다!!

1192 KB

4 ms

696 B

C++14

```
1 #include <cstdio>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 bool check(vector<long long> &a, long long mid, long long m) {
6     int n = a.size();
7     long long cnt = 0;
8     for (int i=0; i<n; i++) {
9         if (a[i]-mid > 0) {
10             cnt += a[i]-mid;
11         }
12     }
13     return cnt >= m;
14 }
15 int main() {
16     int n;
17     long long m;
18     scanf("%d %lld", &n, &m);
19     vector<long long> a(n);
20     long long l=0;
21     long long r=0;
22     for (int i=0; i<n; i++) {
23         scanf("%lld", &a[i]);
24         if (r < a[i]) {
25             r = a[i];
26         }
27     }
28     long long ans = 0;
29     while (l <= r) {
30         long long mid = (l+r)/2;
31         if (check(a, mid, m)) {
32             ans = max(ans, mid);
33             l = mid+1;
34         } else {
35             r = mid-1;
36         }
37     }
38     printf("%lld\n", ans);
39     return 0;
40 }
```

나무 높이

절단기의 높이

절단기 m번

for (int i=0; i<n; i++) {
 if (a[i]-mid > 0) {
 cnt += a[i]-mid;
 }
}

return cnt >= m;

true
false



if (r < a[i]) {
 r = a[i];
}

ans = max(ans, mid);
l = mid+1;

} else {
 r = mid-1;
}

결과

메모리

시간

코드 길이

맞았습니다!!

9044 KB

232 ms

826 B

C++14

```
1 #include <algorithm>
2 #include <iostream>
3 #include <vector>
4 using namespace std;
5 bool possible(vector<int> &a, int c, int x){
6     int cnt = 1;
7     int last = a[0];
8     for (int house : a) {
9         if (house - last >= x) {
10             cnt += 1;
11             last = house;
12         }
13     }
14     return cnt >= c;
15 }
16 int main() {
17     int n, c;
18     cin >> n >> c;
19
20     vector<int> a(n);
21     for (int i=0; i<n; i++) {
22         cin >> a[i];
23     }
24
25     sort(a.begin(), a.end());
26
27     int l = 1;
28     int r = a[n-1] - a[0];
29
30     int ans = l;
31     while(l <= r) {
32         int mid = (l+r)/2;
33         if (possible(a, c, mid)) {
34             if (ans < mid) {
35                 ans = mid;
36             }
37             l = mid + 1;
38         } else {
39             r = mid - 1;
40         }
41     }
42
43     cout << ans << '\n';
44     return 0;
45 }
```

정답(위)리
공유기 개수
가장 인접한
두 공유기 사이
거리

결과

메모리

시간

코드 길이

맞았습니다!!

2380 KB

48 ms

804 B

C++14

```
1 #include <stdio>
2 #include <vector>
3 #include <cstring>
4 using namespace std;
5 vector<pair<int,int>> a[10001];
6 bool c[10001];
7 int n,m;
8 int st, ed;
9 bool go(int node, int limit){
10     if (c[node]) return false;
11     return false;
12 }
13 c[node] = true;
14 if (node == ed) {
15     return true;
16 }
17 for (auto &p : a[node]) {
18     int next = p.first;
19     int cost = p.second;
20     if (cost >= limit) {
21         if (go(next, limit)) {
22             return true;
23         }
24     }
25 }
26 return false;
27 }
28 int main() {
29     scanf("%d %d",&n,&m);
30     while (m--) {
31         int x,y,z;
32         scanf("%d %d %d",&x,&y,&z);
33         a[x].push_back(make_pair(y,z));
34         a[y].push_back(make_pair(x,z));
35     }
36     scanf("%d %d",&st,&ed);
37     int left, right;
38     left = 1;
39     right = 1000000000;
40     int ans = 0;
41     while (left <= right) {
42         int mid = left + (right-left)/2;
43         memset(c,false,sizeof(c));
44         if (go(st, mid)) {
45             ans = mid;
46             left = mid+1;
47         } else {
48             right = mid-1;
49         }
50     }
51     printf("%d\n",ans);
52     return 0;
53 }
```

인장배열

이동하는 중량

node → next
cost

C++14

```
1 #include <cstdio>
2 #include <cmath>
3 #include <algorithm>
4 using namespace std;
5 int main() {
6     double x, y, c;
7     while (scanf("%lf %lf %lf",&x,&y,&c) == 3) {
8         double left = 0;
9         double right = min(x, y);
10        for (int k=0; k<10000 k++) {
11            double mid = (left + right) / 2.0;
12            double d = mid;
13            double h1 = sqrt(x*x-d*d);
14            double h2 = sqrt(y*y-d*d);
15            double h = (h1*h2)/(h1+h2);
16            if (h > c) {
17                left = mid;
18            } else {
19                right = mid;
20            }
21        }
22        printf("%.3lf\n",left);
23    }
24    return 0;
25 }
```

1
21000

결과	메모리	시간	코드 길이
맞았습니다!!	1128 KB	0 ms	629 B

C++14

```

1 #include <iostream>
2 #include <iomanip>
3 #include <cmath>
4 using namespace std;
5 double dist(double x1, double y1, double z1, double x2, double y2, double z2) {
6     return sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1)+(z2-z1)*(z2-z1));
7 }
8 int main() {
9     double x1,y1,z1,x2,y2,z2,x3,y3,z3;
10    cin >> x1 >> y1 >> z1 >> x2 >> y2 >> z2 >> x3 >> y3 >> z3;
11    double dx = x2-x1;
12    double dy = y2-y1;
13    double dz = z2-z1;
14    double left = 0.0;
15    double right = 1.0;
16    double m = 0;
17    while (true) {
18        if (abs(right-left) < 1e-9) {
19            m = (left+right)/2;
20            break;
21        }
22        double m1 = left + (right-left)/3;
23        double m2 = right - (right-left)/3;
24        double m1x = x1 + m1*dx;
25        double m1y = y1 + m1*dy;
26        double m1z = z1 + m1*dz;
27        double m2x = x1 + m2*dx;
28        double m2y = y1 + m2*dy;
29        double m2z = z1 + m2*dz;
30        double d1 = dist(m1x,m1y,m1z,x3,y3,z3);
31        double d2 = dist(m2x,m2y,m2z,x3,y3,z3);
32        if (d1 > d2) {
33            left = m1;
34        } else {
35            right = m2;
36        }
37    }
38    double x0 = x1 + m*dx;
39    double y0 = y1 + m*dy;
40    double z0 = z1 + m*dz;
41    double ans = dist(x0,y0,z0,x3,y3,z3);
42    cout << fixed << setprecision(10) << ans << '\n';
43    return 0;
44 }

```

Handwritten notes and diagrams:

- A line segment is drawn with endpoints labeled (x_1, y_1, z_1) and (x_2, y_2, z_2) . A point (x_3, y_3, z_3) is shown to the right. A perpendicular line segment is drawn from (x_3, y_3, z_3) to the line segment, meeting it at a point labeled m . The distance from (x_1, y_1, z_1) to m is labeled $left$, and the distance from m to (x_2, y_2, z_2) is labeled $right$. The distance from (x_3, y_3, z_3) to m is labeled ans .
- A circle around the `if (abs(right-left) < 1e-9)` condition, with a note 10^{-9} next to it.
- A circle around the `if (d1 > d2)` condition, with a note $0-1$ next to it.
- A circle around the `m` variable in the `double m` declaration.
- A circle around the `x0`, `y0`, and `z0` variables in the `double x0 = x1 + m*dx;` line.

결과

메모리

시간

코드 길이

맞았습니다!!

2000 KB

0 ms

1268 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 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.