SRM 638

Added by vexorian , last edited by vexorian on Nov 23, 2014 (view change)
Labels: (None) EDIT

Single Round Match 638
Wednesday, November 3rd, 2014

Archive
Match Overview
Discuss this match

Match summary

The Problems

NamingConvention | NarrowPassage2Easy | CandleTimerEasy | ShadowSculpture | NarrowPassage2 | CandleTimer

CandleTimer

Rate It Discuss it

Used as: Division One - Level Three:

Value	800
Submission Rate	4 / 400 (1.00%)
Success Rate	3 / 4 (75.00%)
High Score	**lyrically** for 456.13 points (29 mins 55 secs)
Average Score	399.28 (for 3 correct submissions)

Two leaves

A good starting point is to do an analysis of the flames' behavior that amounts to what we did in the division 2 version. The final point will either be a single node or a point between two nodes. In both situations, the point will be the maximum distance between the point and some leaves. But with further analysis, you will notice that we only need two leaves to determine this distance. This is clear in case of the analysis we used to find the time at which two flames meet inside an edge, this time definitely depends only on the distances of the end-points of the edge to two distinct flames. Maybe there are multiple flames that meet at some intermediary point, but this doesn't change our rule.

The only situation in which we do not have two flames meeting at some point is when there is only one flame reaching a leaf of the tree.

From this we can conclude, there are two possibilities for the measured duration and the point at which this duration is attained:

- The final point is a leaf v. The measured time is the distance between another leaf u and v.
- The final point is determined by the furthest point between two leaves u and v. The measured time is this distance. Note that the final point may be a node or a point inside an edge.

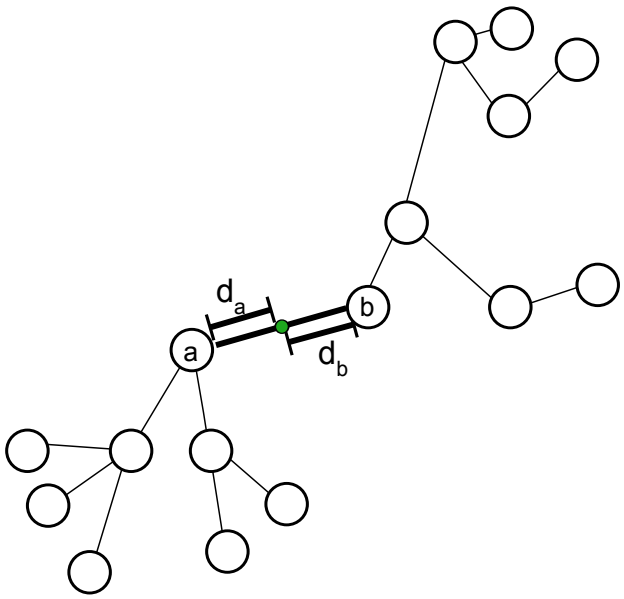
Note that there may always be other leaves that are also set on fire initially. But the final point depends on at most two leaves.

This means there are $O\left(t^2\right)$ different end point / time combinations. However, some of these are invalid (It is not possible to have that end point / time combination because any consistent combination of leaves will yield a different final time than we wanted).

Is it valid?

The key to solve this problem is to be able to answer the question: Given a final point, and a wanted duration, is it possible to pick leaves in such a way that the duration is the one measured?

We will assume the version in which the final point is inside an edge, if we represent this point by the distances from the edge's nodes and the point. The node version can be reduced to the edge version: The distance from one of the edge's endpoints to the final point is 0.



Imagine we wanted the measured time to be l, and the final point is inside the edge between a and b. The distance from a to the final points is d_a and the distance from b to the final point is d_b. There are two subtrees, a and b, in both sub-trees, we want to pick some leaves so that the point is the furthest point starting from the picked set of leaves and this distance is l. For one subtree, we want a to be the furthest node to the picked leaves and the maximum distance should be $l - d_a$. The other subtree is similar, b should be the furthest node from the leaves and the distance $l - d_b$. We will now worry about a single subtree: Imagine we had the subtree root x and we wanted to pick some of the leaves as starting points so that x is the furthest point from those leaves and the maximum distance is w.

If the distance between a leaf and x is too small (smaller than w) then we should not pick that leaf as a starting point. Because then the distance from a picked leaf to x would be smaller than w.

If the distance between a leaf and x is at least as large as w, then picking it is possible. To answer the question of which ones we should pick, the simplest way is to understand that our objective is to make all of the possible points in which flames end, end as soon as possible. We want their end time to be small than or equal to w. The key to solve this is to understand that the more leaves we pick, the smallest the minimum distance from each of the points other than x to a picked leaf will become. So if we pick all of the available leaves, we will minimize all distances other than x. If even when picking all of the available leaves, there is a point other than x with a distance larger than w, then the final point / time pair is unattainable.

Code

We try all $O\left(n^2\right)$ final point / time pairs. For each of them, we pick as many leaves as possible for the initial set of leaves (all leaves such that their distance to x is at least w). If the maximum distance between leaves and every point in the tree is w, then w is a possible time to measure. We can find out the maximum distance by reusing the idea from division 2.

```
const int INF = 1001001001;

int N;
vector<int> G[1010];
int D[201][201];

int dist[1010];
vector<int> leaves;


bool check(int a, int da, int b, int db, int l)
{
// l is the wanted distance
// a is a vertex with distance da to the intended final point
// b is the other vertex with distance db to the intended final point

    set<tuple<int,int>> q;
    fill(dist, dist + N, INF);

    for (int u : leaves) {
        if (D[u][a] <= D[u][b]) {
            if (D[u][a] + da >= l) {
                dist[u] = 0;
                q.emplace(0, u);
            }
        } else {
            if (D[u][b] + db >= l) {
                dist[u] = 0;
                q.emplace(0, u);
            }
        }
    }
    for (auto tup : q) {
```

Alternative solutions and additional comments.

<Place your comments here>

By **vexorian**

TopCoder Member

5 Comments | Add Comment