Analysis: Space Emergency

In this problem, you have a series of edges to follow and you're given the option of shortening up to **L** of the edges by a factor of two. There's one catch: the shortening takes some time to take effect.

Fortunately, all possible shortenings take the same amount of time, so we can divide our edges into three categories:

1. Edges that your flagship will pass before any speed boosters can be built.
2. Edges whose speed boosters would finish **while** your flagship is moving along the edge.
3. Edges whose speed boosters can be built before your flagship gets there.

Group 2 only has zero or one members, because we know exactly where your flagship will be when the speed boosters finish building.

Now we can decide how useful it is to build a speed booster on each edge. We never want to build speed boosters on group 1, because they won't help. For each edge in group 3, the benefit is length/2. For the edge in group 2, the benefit is (distance to the end of the edge from where your flagship will be when the speed booster finishes building)/2.

We choose the **L** most beneficial edges to build on; if there aren't **L** beneficial edges, we'll stop when we've built on all the beneficial edges. Then we compute the total benefit, subtract it from the total time to traverse the whole path, and voilÃ ! The answer.

The way in which we specify edges in this problem is periodic: if the input is very large, then there must be a lot of edges with the same lengths. A clever solution could take advantage of the periodic nature to run very quickly, but the limits were small enough that this shouldn't be necessary; we actually specified the input in this way because it would make the input file smaller, not because we wanted to test that particular skill.