* 1. Change number n to base 2, it will be shown as which is 0 or 1.

=

So we only need to calculate matrices, each one is the square of the previous one.

The total steps are, time complexity is



Using 3 real numbers multiplications and 5 real numbers additions.

Using 2 real numbers multiplications and 3 real numbers additions

Using 3 real numbers multiplications we could calculate

Using 2 real numbers multiplications we could calculate

So using 5 multiplications only.

2. Let the edge length as the node value

int transform(Node<Integer> root) {

if (root == null) {

return 0;

}

int leftValue = transform(root.getLeft());

int rightValue = transform(root.getRight());

int maxValue = Math.max(leftValue, rightValue);

if (leftValue != 0) {

root.getLeft().setValue((int) root.getLeft().getValue() + maxValue - leftValue);

}

if (rightValue != 0) {

root.getRight().setValue((int) root.getRight().getValue() + maxValue - rightValue);

}

return maxValue + root.getValue();

}

2. 1. Set Loololong as START
   2. Set first house start from START as FIRST;
   3. Go straight until reaching the position which is 5 km far away from FIRST;
   4. Set a base station on this position;
   5. Move forward 5km and set current position as START
   6. Repeat step 2 to 5 until reaching Goolagong;
3. 1. Find the people in the people list whose known number or unknown number is less than 5.
   2. Remove this one from people list and Pairs list.
   3. Repeat finding the people and remove him/her until everyone’s known number and unknown number is greater than or equal to 5.
4. 1. If start by connecting the closest pair of black and a white dot, Connection A would be one of the possibility. But the total length is 4 which is larger than the optimal length 2 as connection B.

B

A

* 1. The optimal solution:

Select the left most white node and left most black node.

Connect and ignore them.

Repeat

1. Create a ***List*** which length is n.

For task i = 1 to n:

If from ***List[0]*** to ***List[i-1]*** , one or more are empty:

Set into the right most one.

Else:

If is less than all values:

Add to ***totalPenalty***

Else:

Set into the one has minimum value.

Add that minimum value to ***totalPenalty***