CMIMC

Theoretical Computer Science Round

Instructions

1. Do not look at the test before the proctor starts the round.
2. This test consists of three problems, which require proofs, to be solved within a time frame of 60 minutes. There are 300 points total.
3. No computational aids other than pencil/pen are permitted.
4. Answers should be written and clearly labeled on sheets of blank paper. Each numbered problem should be *on its own sheet*. If you have multiple pages, number them as well (e.g. 1/3, 2/3).
5. Write your team name on the upper-right corner and the problem and page number of the problem whose solution you are writing on the upper-left corner on each page you submit. Papers missing these will not be graded. Problems with more than one submission will not be graded.
6. Write legibly. Illegible handwriting will not be graded.
7. If you believe that the test contains an error, submit your protest to the 2024 CMIMC discord.

The Compass and The Edge

Partial points may be awarded for progress in the constructions below. Partial points may be deducted for logical flaws or lack of rigor in proofs. To get full points, you must demonstrate that your construction works. **If there is a bound, you must show this construction always achieves at most your claimed worst-case scenario.** Please write each part of each problem on a separate page with the problem part in it, e.g. 1. a).

1.Allan, intrigued by his geometry class, has decided to purchase a compass and a ruler. However, his ruler ended up having no markings! To start, Allan used his ruler, now straightedge, to draw a short segment that he confidently calls an “inch”.

Allan wants to recreate a ruler, but with his own custom markings. As they are short on time, they want to minimize the number of compass and straightedge operations they will use.

Allan is able to use the construction materials as follows:

- A straightedge can draw a line through two (possibly arbitrary) points. Allan can also stop drawing prematurely to simply draw a segment. This will take 1 step.
- A compass is able to draw a circle given three (possibly not distinct) points, O, A, B. Then Allan can draw a circle with radius length AB with the center at O. This will take 1 step.
- Allan is also able to draw a (possibly arbitrary) point or find the intersection of two objects. This will take 0 steps.

For his ruler, Allan wants to construct a segment of $\frac{1}{2^n}$ inches for a positive integer n.

Scoring

a) A proper construction for a segment of length $\frac{1}{2^n}$ will earn

- 5 points for a construction.

Instead of powers of 2, Allan is interested in any integer and wants to first construct a segment of length $\frac{1}{n}$ ”inches”. More precisely, given a segment of length 1, give a process that can construct/draw a segment of length $\frac{1}{n}$ for any integer n.

Scoring

b) A proper construction in N steps will earn

- 1 point for a construction
- 5 points for $N \leq n$
- 15 points for $N \leq 2\log_2 n + c$ for a constant c.
- 25 points for $N \leq \log_2 n + c$ for a constant c.

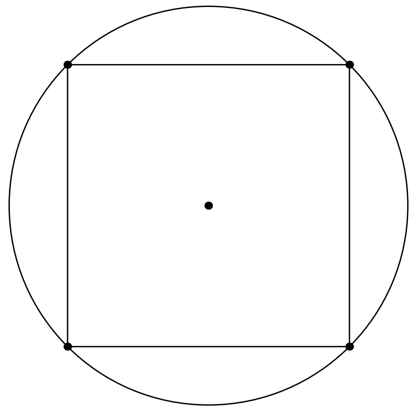
Now that Allan constructed a segment of length $\frac{1}{n}$, he still needs to complete marking his ruler. To generalize, he wants to construct *any* rational amount of ”inches”. Give a process that can construct a segment of length $\frac{p}{q}$ ”inches” where p and q are coprime positive integers.

Scoring

c) A proper construction in N steps will earn

- 1 point for a construction
- 5 points for $N \leq \log_2 p + \log_2 q$.

Allan now is eager to draw something more interesting, namely a square! Already having a circle (and center) in front of him, he aims to draw this square so that it is inscribed in the circle, as seen below.



This doesn't sound too bad, but Allan only has his straightedge with him! Given this circle and center, help Allan draw the inscribed square.

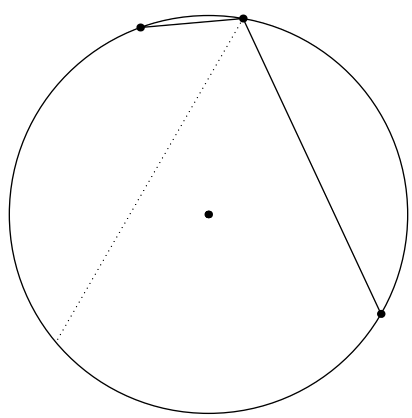
After noticing a few of his line segments were not perfectly straight, Allan hurriedly grabs his construction tools and his circle and wants to redo this construction. This time, Allan only has a compass! This time, construct the vertices of the square that would be inscribed in the circle.

Scoring

d) A proper construction will earn

- 15 points for a construction using only a compass.
- 10 additional points for a construction using only a straightedge, given a circle with its center.

Frustrated by the last endeavor, Allan wants to do something simpler: bisect an angle. Given an inscribed angle in a circle with a center, help Allan draw the angle bisector, but without his compass.



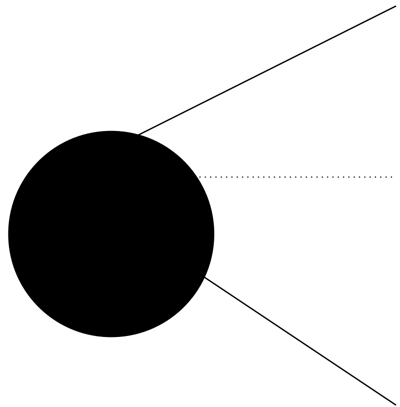
Scoring

e) A proper construction will earn

- 15 points for a construction using only a straightedge, given a circle with its center.

Allan, still frustrated by the previous problems, wants to construct something seemingly really easy with his compass AND straightedge: a line that intersects at the same point as two given lines. As Allan pulls out his straightedge, he spills ink over the intersection point! Now, any marks he makes over the ink is not visible.

More formally, given two nonparallel lines AB and CD, shown in solid line construct a third line EF (dashed) where AB, CD, and EF are concurrent.



Scoring

f) A proper construction will earn

- 15 points for a construction using a compass and straightedge
- 10 additional points for a construction using only a straightedge

Where the Tree At?

Andrew C. has a network of 10^{10} nodes, with some pairs of them connected. All he knows is that the graph of the connections between pairs is a tree (there are no cycles and any two points in the graph have a path between them of connected nodes) and has max degree of 10. He wants to figure out what his network looks like, but the only tool he has is to figure out the length of the path between two nodes. This algorithm should be optimized to minimize the worst case total number of queries. Submit an algorithm, and an M value such that your algorithm will run in M queries or fewer for any tree.

Scoring

- 100 pts for $M = 1.1 \times 10^{12}$
- 90 pts for $M = 1.12 \times 10^{12}$
- 75 pts for $M = 3.3 \times 10^{12}$
- 65 pts for $M = 3.5 \times 10^{12}$
- 45 pts for $M = 2.49 \times 10^{19}$
- 30 pts for $M = 2.5 \times 10^{19}$
- 10 pts for some $M < \binom{10^{10}}{2}$
- 5 pts for $M = \binom{10^{10}}{2}$
- 1 pt for some finite value

Wanted: Luigi

After stealing a star from Bowser, Luigi is on the run from Bowser's army! Luigi has disappeared into a crowd of Marios and Yoshis on an $X \times Y$ board.

On this board, some of the squares contain a copy of Mario, some of the squares contain a copy of Yoshi, and exactly one square contains Luigi. Each second, every Mario moves one square to the right, every Yoshi moves one square down, and Luigi can choose whether to move one square right or down. When a character moves off the edge of the board, they loop around to the opposite edge.

M	MY**L**		M
Y	MY		Y
	MY	M	MY

M	MY	M	Y
	Y**L**	M	
MY	Y	M	MY

Y	MY	M	MY
	Y	**L**	MY
M	MY		M

Figure 1: A sample 3×4 board across 3 seconds. M is Mario, Y is Yoshi, and **L** is Luigi.

Bowser has deployed a genius algorithmic StemKoopa to retrieve the star, which has been hidden at **Luigi's initial location**. StemKoopa can see each Mario and Yoshi normally, but the square containing Luigi just appears to contain Mario and Yoshi. Their goal is to quickly determine Luigi's initial location (or report to Bowser that his initial location can never be found). They have coded up the optimal algorithm to do so.

M	MY		M
Y	MY		Y
	MY	M	MY

M	MY	M	Y
	MY	M	
MY	Y	M	MY

Y	MY	M	MY
	Y	MY	MY
M	MY		M

Figure 2: The same sample boards from StemKoopa's perspective

Each second, if StemKoopa definitively knows Luigi's initial location or definitively knows they'll never be able to derive it (based on what they have seen so far), they will report as such to Bowser. Otherwise, they will wait another second for Luigi to move to see if he gives himself away. You may assume StemKoopa's algorithm is optimal and comes to its conclusion at the earliest possible second (possibly the 0th second), and that Luigi will move optimally.

Task

Your goal is to relay crucial information to Peach on how long it will be until StemKoopa finds the star. In particular, given an initial state of the board (including Luigi's location), describe an algorithm that outputs T, the maximum number of seconds Luigi can keep his initial location a secret from StemKoopa.

To make sure Peach gets the information in time, your algorithm should run in $O(XY)$ time. It may be helpful to consider the problem from StemKoopa's perspective (who can't see Luigi), and how doing so induces an $O(X^2Y^2)$ solution.

Scoring

For full credit, please indicate which set of assumptions your algorithm makes, and justify the correctness of your algorithm.

- 10 points for an algorithm that assumes no square starts with more than one character (Mario, Yoshi, Luigi)
- 20 points for an algorithm that assumes Luigi begins on a square by himself (i.e. no Marios or Yoshis on his square)
- 50 points for an algorithm that assumes Luigi begins on a square with at most one other character on his square, and that $\gcd(X, Y) = 1$
- 75 points for an algorithm that assumes $\gcd(X, Y) = 1$
- 100 points for an algorithm that makes no further assumptions

Giving a worse-than $O(XY)$ solution for any part gets at least 35% of the credit for that part. If the solution can easily be modified to meet the time bounds, the deduction will be much less.