Solutions

Problem H

Problem I

Problem D

Problem J

Problem A

Problem F

Problem B

Problem C

Problem K

Problem E

NWERC 2008
Solutions to the problems

The Jury

Utrecht University
The Netherlands

acm International Collegiate Programming Contest

IBM event sponsor

H - Matchsticks

- For largest number: use lots of 1s
- Start with 1 or 7 (depending on n mod 2)
- For smallest number: use lots of 8s
- Start with 108,188,200,208,288,688,888 (depending on n mod 7)
- Small numbers can be tricky: brute force them

H - Matchsticks

- For largest number: use lots of 1s
- Start with 1 or 7 (depending on n mod 2)
- For smallest number: use lots of 8s
- Start with 108,188,200,208,288,688,888 (depending on n mod 7)
- Small numbers can be tricky: brute force them

- Statistics: 145 submissions, 47 correct (EVERYONE!), first 20 minutes

Solutions

Problem H
Problem I
Problem D
Problem J
Problem A
Problem F
Problem B
Problem C
Problem K
Problem E

I - Rafting

- The answer is the minimum distance between the two polygons
- Calculate distances between points and line segments to find it

I - Rafting

- The answer is the minimum distance between the two polygons
- Calculate distances between points and line segments to find it

- Statistics: 67 submissions, 30 correct, first 96 minutes

D - Disgruntled Judge

- Loop over A and B and generate the sequence
- Break as soon as it doesn't match
- That's all
- Note: number theory gives much faster solutions

Solutions

Problem H

Problem I

Problem D

Problem J

Problem A

Problem F

Problem B

Problem C

Problem K

Problem E

D - Disgruntled Judge

- Loop over A and B and generate the sequence
- Break as soon as it doesn't match
- That's all
- Note: number theory gives much faster solutions

- Statistics: 61 submissions, 28 correct, first 31 minutes

J - Shuffle

- Count how many different songs are in the intervals of length *s*.
- Update for a next interval in O(1) time by adding and removing one song
- Then check which positions are valid

J - Shuffle

- Count how many different songs are in the intervals of length s.
- Update for a next interval in O(1) time by adding and removing one song
- Then check which positions are valid

- Statistics: 116 submissions, 21 correct, first 52 minutes

A - Mobile

- All weights at a certain level must have the same weight
- All weights one level higher must have twice that weight, and so on
- Calculate all $2^{-\text{depth}} \times$ weight
- Use 64-bit integers for that
- Find the most recurring one (e.g. by sorting first)

Solutions

Problem H

Problem I

Problem D

Problem J

Problem A

Problem F

Problem B

Problem C

Problem K

Problem E

A - Mobile

- All weights at a certain level must have the same weight
- All weights one level higher must have twice that weight, and so on
- Calculate all $2^{-\text{depth}} \times \text{weight}$
- Use 64-bit integers for that
- Find the most recurring one (e.g. by sorting first)

- Statistics: 43 submissions, 12 correct, first 167 minutes

F - Sculpture

- Compress coordinates to 0..100
- Draw the boxes in a 100 × 100 × 100 array
- Flood fill the outer region
- Count the area and volume by using the original coordinates

F - Sculpture

- Compress coordinates to 0..100
- Draw the boxes in a 100 × 100 × 100 array
- Flood fill the outer region
- Count the area and volume by using the original coordinates

- Statistics: 18 submissions, 7 correct, first 130 minutes

Solutions

Problem H
Problem I
Problem D
Problem J
Problem A
Problem F
Problem B
Problem C
Problem K
Problem E

B - Equivalences

- Find strongly connected components with a DFS (see your favorite algorithm book for that)
- Count how many components have in-degree 0 and out-degree 0
- Maximum of these is the answer
- Corner case: if there is a single s.c.c. , the answer is 0

B - Equivalences

- Find strongly connected components with a DFS (see your favorite algorithm book for that)
- Count how many components have in-degree 0 and out-degree 0
- Maximum of these is the answer
- Corner case: if there is a single s.c.c. , the answer is 0

- Statistics: 58 submissions, 6 correct, first 149 minutes

C - Cat vs Dog

- Make a bipartite graph with cat lovers and dog lovers as vertices
- Add an edge if their votes are incompatible
- Problem now is: find minimum vertex cover
- Equivalent to maximum matching for bipartite graphs

C - Cat vs Dog

- Make a bipartite graph with cat lovers and dog lovers as vertices
- Add an edge if their votes are incompatible
- Problem now is: find minimum vertex cover
- Equivalent to maximum matching for bipartite graphs

- Statistics:20 submissions, 5 correct, first 85 minutes

Solutions

Problem H

Problem I

Problem D

Problem J

Problem A

Problem F

Problem B

Problem C

Problem K

Problem E

K - Videopoker

- Before processing testcases: generate all poker hands and rankings
- For a testcase, loop over all hands
- Count how many cards you have to change for a hand
- Average the results and calculate the expectation value for each change

K - Videopoker

- Before processing testcases: generate all poker hands and rankings
- For a testcase, loop over all hands
- Count how many cards you have to change for a hand
- Average the results and calculate the expectation value for each change

- Statistics: 1 submission, ?? correct, first ??? minutes

E - Easy Climb

- Only heights of the form $h_i + nd$ are relevant (so n^2 heights)
- Dynamic programming: calculate best[x][h]
- Use monotonocity property to update in amortized O(1) time
- This gives an $O(n^3)$ algorithm

E - Easy Climb

Solutions

Problem H

Problem I

Problem D

Problem J

Problem A

Problem F

Problem B

Problem C

Problem K

Problem E

- Only heights of the form $h_i + nd$ are relevant (so n^2 heights)
- Dynamic programming: calculate best[x][h]
- Use monotonocity property to update in amortized O(1) time
- This gives an $O(n^3)$ algorithm

- Statistics: 0 submissions, 0 correct, first 0 minutes