TWGSS
OI Team

Operation Imaginary II

Editorial

Typhoon Kompasu

TW2221

Problem Statement

You have a ruler and a compass.

Draw a N-units segment, given two points separated by 1 unit.

Trivial Solution

Induction!

O(N)

Draw a line joining initial

Repeatedly draw circle …

C = 2000

Score: 20.000

Slightly optimized (C ~ 130)

Set B = block size. We find 0…2B – 1, then extend it by B each time.

B = 25: C = 129

B = 30: C = 125

B = 31: C = 125

B = 32: C = 125

B = 33: C = 125

B = 34: C = 125

B = 35: C = 126

B = 40: C = 129

B = 45: C = 133

B = 50: C = 139

2B + N / B ~ 2B^2 = N => B ~ 32

Score: 48.000 (C = 130)

Slightly optimized (C = 125)

C = 125 for this approach

Score: 49.298

Further optimized (C = 123)

We can skip the later part of first step if it is not used anyways

C = 32: 123

Score: 49.817

Binary! (C = 22)

Why not?

Repeat the block…

Define plot(a, b): plot 2b – a from a and b

0, 1 -> 2

1, 2 -> 3

1, 3 -> 5

3, 5 -> 7

C = 22

Score: 81

10111

01111

00111

00101

00011

00010

00001

00000

~ O(2#bits = 1) = 2 * 11 = 22 (N = 1023)

Further into the hole

Is there any solution better than C = 22?

‘Any points’

Is there any exploit on this?

Hard bound (C >= 12)

Note that theoretical minimum = 1 + ceil(lg N) = 12 (N = 2000)

So the possibility is not large

Further into the hole (C = 21)

10111

01111

00111

00101

00011

00010

00001

00000

Note that this is maximum when binary of N is mostly consists of 1

Further into the hole (C = 21)

Hardcode for N = 1023:

0, 1, 2, 4, …, 1024

Answer(1, 1024)

C = 21

Score: 82.5

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu\solution.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

Project Classes Debug

co_geom.h test.cpp kompasu.cpp kompasu.h grader.cpp solution.cpp grader_multiple.cpp

```
19      pair<int, int> X = intersections(c1, c2);
20      int perp = drawLine(X.first, X.second);
21      pair<int, int> X2 = intersections(l, perp);
22      p[(a + b) / 2] = X2.first;
23  }
24
25  void kompasu(int N) {
26      l = drawLine(1, 2);
27      p[0] = 1, p[1] = 2;
28      int c = 0;
29      if(N == 1023 || N == 1535 || N == 1791 || N == 1919 || N == 1983){
30          plot(0, 1);
31          plot(0, 2);
32          plot(0, 4);
33          plot(0, 8);
34          plot(0, 16);
35          plot(0, 32);
36          plot(0, 64);
37          plot(0, 128);
38          plot(0, 256);
39          plot(0, 512);
40          plot(0, 1024);
41          if(N == 1023){
42              answer(p[1], p[1024]);
43          }else if(N == 1535){
44              plot(512, 1024);
45              answer(p[1], p[1536]);
46          }else if(N == 1791){
47              plot(256, 1024);
48              answer(p[1], p[1792]);
49          }else if(N == 1919){
50              plot(128, 1024);
51              answer(p[1], p[1920]);
52          }else if(N == 1983){
53              plot(64, 1024);
54              answer(p[1], p[1984]);
55          }
56      }else{
57          for(int i = 0; (1 << i) < N; i++){
58              plot(c, c + (1 << i));
```

選取 命令提示字元

```
>>> exit()

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution
solution.cpp: In function 'void mid(int, int)':
solution.cpp:21:24: error: redeclaration of 'std::pair<int, int> X'
   21 |         pair<int, int> X = intersections(l, perp);
      |                        ^
solution.cpp:19:24: note: 'std::pair<int, int> X' previously declared here
   19 |         pair<int, int> X = intersections(c1, c2);
      |                        ^

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
OK
21
1535 1791 1919 1983

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
WA
The distance between the two points is 6.0000000 instead of 1535.

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution
```

Compiler (17) Resources Compile Log Debug Find Results Close

Line	Col	File	Message
		C:\Users\User\Documents\code\problemsetting\TW...	In member function 'point line::unit(long double)':
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected '}' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] could not convert '{{{{{(line*)this)->line::b * l) / std::sqrt({{{{{(line*)this)->line::a * ((line*)this)->line::a) + {{{(line*)this)->line::b * ((line*)this)->line::b))))}' from '<brace-enclosed initializer list>' to 'point'
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected ';' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected primary-expression before ')' token
		C:\Users\User\Documents\code\problemsetting\TW...	In function 'point intersection(line, line, int)':
111	95	C:\Users\User\Documents\code\problemsetting\TW222	[Error] 'WA' was not declared in this scope

Line: 54 Col: 32 Sel: 0 Lines: 66 Length: 1371 Insert

CH 速 26°C 晴時多雲 23:01 25/5/2022

Further (C = 20)

Hardcode for these cases with many ‘1’s!

N = 1535, 1791, 1919, 1983 (bounded by N = 2000, or actually more)

Note that just use (1, 2^M, 2^N) is sufficient

C = 20

Score: 84

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu\solution.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

Project Classes Debug

co_geom.h test.cpp kompasu.cpp kompasu.h grader.cpp solution.cpp grader_multiple.cpp

```
19      pair<int, int> X = intersections(c1, c2);
20      int perp = drawLine(X.first, X.second);
21      pair<int, int> X2 = intersections(l, perp);
22      p[(a + b) / 2] = X2.first;
23  }
24
25  void kompasu(int N) {
26      l = drawLine(1, 2);
27      p[0] = 1, p[1] = 2;
28      int c = 0;
29      if(N == 1023 || N == 1535 || N == 1791 || N == 1919 || N == 1983){
30          plot(0, 1);
31          plot(0, 2);
32          plot(0, 4);
33          plot(0, 8);
34          plot(0, 16);
35          plot(0, 32);
36          plot(0, 64);
37          plot(0, 128);
38          plot(0, 256);
39          plot(0, 512);
40          plot(0, 1024);
41          if(N == 1023){
42              answer(p[1], p[1024]);
43          }else if(N == 1535){
44              plot(512, 1024);
45              answer(p[1], p[1536]);
46          }else if(N == 1791){
47              plot(256, 1024);
48              answer(p[1], p[1792]);
49          }else if(N == 1919){
50              plot(128, 1024);
51              answer(p[1], p[1920]);
52          }else if(N == 1983){
53              plot(64, 1024);
54              answer(p[1], p[1984]);
55          }
56      }else{
57          for(int i = 0; (1 << i) < N; i++){
58              plot(c, c + (1 << i));
```

命令提示字元

```
solution.cpp:19:24: note: 'std::pair<int, int> X' previously declared here
   19 |         pair<int, int> X = intersections(c1, c2);
      |                        ^

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
OK
21
1535 1791 1919 1983

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution.cpp -o grader_mul

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
WA
The distance between the two points is 6.0000000 instead of 1535.

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution.cpp -o grader_mul

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
OK
20
767 895 959 991 1007 1015 1019 1021 1022 1279 1407 1471 1503 1519 1527 1531 1533 1534 1663 1727 1759 1775 1783 1787 1789 1
790 1855 1887 1903 1911 1915 1917 1918 1951 1967 1975 1979 1981 1982 1999

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>
```

Compiler (17) Resources Compile Log Debug Find Results Close

Line	Col	File	Message
		C:\Users\User\Documents\code\problemsetting\TW...	In member function 'point line::unit(l
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected '}' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] could not convert '{{{{{line*)this}
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected ';' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected primary-expression before ')' token
		C:\Users\User\Documents\code\problemsetting\TW...	In function 'point intersection(line, line, int)':
111	95	C:\Users\User\Documents\code\problemsetting\TW222	[Error] 'WA' was not declared in this scope

Line: 54 Col: 32 Sel: 0 Lines: 66 Length: 1371 Insert

CH 速 26°C 晴時多雲 23:03 25/5/2022

Can it be further?

As you see there are a lot of equal cases

~~Hard code each then C can be reduced to 19~~

Too time consuming!

Time to change our approach

Is there any general method to tackle the case that there are many '1's?

C = 15

If there are two ‘1’ continuous, we move left by 1

We can ignore smaller bits recursively

C = 15

Score: 91.5

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu\solution.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

Project Classes Debug

co_geom.h test.cpp kompasu.cpp kompasu.h grader.cpp solution.cpp grader_multiple.cpp

```cpp
#include "kompasu.h"

#include <bits/stdc++.h>

using namespace std;
map<int, int> p;

int l;

void plot(int a, int b){
    int c = drawCircle(p[b], p[a]);
    pair<int, int> X = intersections(l, c);
    p[b * 2 - a] = (a < b ? X.second : X.first);
}

void mid(int a, int b){
    int c1 = drawCircle(p[b], p[a]);
    int c2 = drawCircle(p[a], p[b]);
    pair<int, int> X = intersections(c1, c2);
    int perp = drawLine(X.first, X.second);
    pair<int, int> X2 = intersections(l, perp);
    p[(a + b) / 2] = X2.first;
}

void kompasu(int N) {
    l = drawLine(1, 2);
    p[0] = 1, p[1] = 2;
    int c = 0;
    for(int i = 0; (1 << i) < (N + c); i++){
        if(((N + c) & (1 << i))){
            plot(c + (1 << i), c);
            c -= (1 << i);
        }else{
            plot(c, c + (1 << i));
        }
    }
    answer(p[c], p[N + c]);
}
```

命令提示字元

```
C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>g++ grader_multiple.cpp solution.cpp -o grader_mul

C:\Users\User\Documents\code\problemsetting\TW2221 - Typhoon Kompasu>grader_mul
OK
11
1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 10
48 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072
 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1
097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 112
1 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140 1141 1142 1143 1144 1145
1146 1147 1148 1149 1150 1151 1152 1153 1154 1155 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 11
70 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194
 1195 1196 1197 1198 1199 1200 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215 1216 1217 1218 1
219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 124
3 1244 1245 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260 1261 1262 1263 1264 1265 1266 1267
1268 1269 1270 1271 1272 1273 1274 1275 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290 1291 12
92 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316
 1317 1318 1319 1320 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335 1336 1337 1338 1339 1340 1
341 1342 1343 1344 1345 1346 1347 1348 1349 1350 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 136
5 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389
1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 14
14 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438
 1439 1440 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455 1456 1457 1458 1459 1460 1461 1462 1
463 1464 1465 1466 1467 1468 1469 1470 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485 1486 148
7 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511
1512 1513 1514 1515 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530 1531 1532 1533 1534 1535 15
36 1537 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560
 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575 1576 1577 1578 1579 1580 1581 1582 1583 1584 1
```

Compiler (17) Resources Compile Log Debug Find Results Close

Line	Col	File	Message
		C:\Users\User\Documents\code\problemsetting\TW...	In member function 'point line::unit(l
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected '}' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] could not convert '{{{{{line*}this}
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected ';' before ')' token
77	39	C:\Users\User\Documents\code\problemsetting\TW222...	[Error] expected primary-expression before ')' token
		C:\Users\User\Documents\code\problemsetting\TW...	In function 'point intersection(line, line, int)':
111	95	C:\Users\User\Documents\code\problemsetting\TW222	[Error] 'WA' was not declared in this scope

Line: 21 Col: 28 Sel: 0 Lines: 38 Length: 792 Insert

CH 速 26°C 晴時多雲 23:30 25/5/2022

Final Countdown (C = 12)

Move left move right repeatedly

Theoretical maximum!

C = 12

Expected Score: 100

Circle-circle intersection is never used!

Indeed, this is just a camouflage since it is unnatural to ‘ban’ circle-circle intersection ☺

DVD Logo

TW2222

Problem Statement

A DVD Logo bouncing, starting at (x, y) and initially move (v_x, v_y) per second.

If it will eventually hit the corner, output **YES** on the first line and the time needed on the second line.

Otherwise, output **NO**.

Subtask 1

$v_x = 0$
$1 \leq N, M \leq 1000$

Trivial case work.

- $y = 0$ or M → **YES**
- Otherwise → **NO**

Score: 7

Subtask 2

$v_x = v_y = 1$
$1 \leq N, M \leq 1000$

Simulate.

Remember to check first whether it is initially at corner.

Score: 30

Subtask 3

N and x is divisible by v_x

M and y is divisible by v_y

If v_x is negative, can set $v_x := -v_x$ and $x := n - x$ without changing the answer.

Similar for v_y.

Speed up simulation.

Score: 45

Partial Score - Yes/No only

Extend the TV

We have N | x + tv_x, M | y + tv_y, t is a rational number

So aN = x + tv_x, bM = y + tv_y

t = (aN - x) / v_x

bM = y + (aN - x) / v_x * v_y

bM * v_x – aN * v_y = y * v_x – x * v_y

So YES iff (bv_x, av_y) | y * v_x – x * v_y

Score: 35.6

Subtask 4-5

bM * v_x – aN * v_y = y * v_x – x * v_y

Seems complicated!

But it is just in form of

bX – aY = Z, X = Mv_x, Y = Nv_y, Z = yv_x – xv_y

Since X and Y are nonnegative integers, we may increase X and Y depending on bX – aY > Z or bX – aY < Z

If bX – aY > Z, increase Y by 1

If bX – aY < Z, increase X by 1

If bX – aY = Z, we have find the solution: output t = (aN - x) / v_x

Repeatedly until X and Y are too large (e.g. > 1e6), and output 'NO'

Score: 83 (wrong initial: 28.8)

Subtask 6

Solve bM * v_x – aN * v_y = y * v_x – x * v_y by extended euclidean algorithm

Note that a >= x / N and b >= y / M

Solving it and we are done

Remember to compute division in floating point in the last step

Score: 100

Colourful Highlighter

TW2223

Problem Statement

There are N highlighters

You may do the two type of operations:

moveLeft(i): Move the highlighter currently indexed i to left end.

moveRight(i): Move the highlighter currently indexed i to right end.

Find the minimum number of operations needed, and output any one of the possible sequence of operations.

Subtask 1

Hard code each of the 6 cases for N = 3

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1

Score: 5

Subtask 2

Brute force simulation

Observation 1. You never move a highlighter more than once.

Therefore, you may just choose an order and ‘L’ or ‘R’ and simulate them.

Time Complexity: O(N * N! * 2^N)

Score: 17

Subtask 3

Define L and R the number of times moveLeft() and moveRight() is used respectively

Observation 2: moveLeft() and moveRight() does not affect each other

Observation 3a: If L and R is fixed, if c_i are all unique, then there exist only one possible choice to move the highlighters, from large to small **in value** for L and from small to large **in value** for R.

Just go through all pairs of L, R, and find the order

Time complexity: O(N^3)

Score: 28 (Cumulative: 40)

Subtask 4

~~Observation 3a: If L and R is fixed, if c_i are all unique, then there exist only one possible choice to move the highlighters, from large to small for L and from small to large for R.~~

No longer holds as there are equal values in c_i

Observation 3b: If L and R is fixed, there exist only one **best** choice to move the highlighters, from large to small **in value and index** for L and from small to large **in value and index** for R.

L increase -> R increase, so just have to check O(N) pairs of (L, R)

Remember subtask 1 cases

Time complexity: O(N^2)

Score: 53

Subtask 5

To maintain whether an array is sorted, we may notice that if it is sorted, then there exist a boundary for C = 1 and 2

We maintain the left and right bound of C = 1 and 2 respectively, and update them accordingly

O(1) update when L or R change

After finding L and R, just choose the last L '1' and first R '2' and move them to the left and right respectively.

Time complexity: O(N) or O(N lg N)

Score: 30 (Cumulative: 83)

Subtask 6

Combine idea of subtask 4 and 5

Use a set to maintain the ‘sorted’ sequence in the middle

Increase R until can’t, then increase L

Time Complexity: O(N lg N)

Score: 100

Blossom Tree

TW2224

Problem Statement

東風夜放花千樹。更吹落、星如雨。
寶馬雕車香滿路。鳳簫聲動，玉壺光轉，一夜魚龍舞。
蛾兒雪柳黃金縷。笑語盈盈暗香去。
衆裏尋他千百度。驀然回首，那人卻在，燈火闌珊處。

《青玉案．元夕》

There are N trees in [-P, P], each at position x_i with lightness of l_i

The contribution of light by tree i at position x is max(0, l_i – abs(x – x_i))

Find the dimmest position and the respective lightness

Subtask 1

For N = 1,

Check x = -P and P. Find the dimmer one.

For N = 2,

Check x = -P, P and start/end point of each light segment in the middle.

Always remember to find the one with the largest index

Expected score: 8

Subtask 2 (Solution 1)

Loop over N and P

For each position i = -P, -P+1, …, P:

```
light = 0
for each tree j:
    light += l_j – abs(i – x_j);
update ans
```

Time complexity: O(NP)

Expected Score: 12

Subtask 2 (Solution 2)

Loop over N and I instead

For each tree i:

```
for (j = max(P, x_i – l_i); j <= min(P, x_i + l_i); j++):
    light[j] += l_j – abs(i – x_j)
```

check ans

Score: 12 (Cumulative: 20)

Subtask 3

Special points

Note that only the points –P, P, x[i] – l[i], x[i], x[i] + l[i] are ‘special’

For any other points, we can move left or move right to further minimize the lightness

[add graph]

There are O(3N) special points

Time complexity: O(3N) * O(N) = O(N^2)

Score: 30 (Cumulative: 50)

Subtask 4

Partial-partial sum

Or just sort the indexes and maintain it on the go

[Insert diagram here]

Score: 45 (Cumulative: 83)

Subtask 5

Combine idea of subtask 3 and 4

We may maintain the double partial difference and compute back the lightness of all the special points

Score: 100