

2013 年度も、昨年同様 5 時間で 10 問を出題した。正解数ごとのチーム数を表 1 に示す。また、問題ごとの提出数・正解チーム数・最初の正解時間を表 2 に示す。今年は、全チームが問題 A と問題 B の両方に正解した。最上位チームの正解数は 9 であり全問正解のチームはなかったが、全問題に対して解答プログラムの提出があった。

今年の問題セットは、易しい問題 (A・B)、中程度の問題 (C・D・E)、難しい問題 (F・G・H・I)、大変難しい問題 (J) からなる。結果として正解数ごとのチーム数がうまい具合に分散しており、問題セットの難易度としては適切であったと考えている。また難しい問題の中でも解く順番や解いた問題が分散していた。

表 1: 正解数ごとのチーム数

正解数	10	9	8	7	6	5	4	3	2	1	0
チーム数	0	2	1	3	5	6	13	6	4	0	0

表 2: 問題ごとの結果 (全 40 チーム)

問題	分野	提出数	正解 チーム数	最初の正解 時間 (分)
A	数の集合の列挙	42	40	6
B	シミュレーション	47	40	14
C	領域の数え上げ	74	31	28
D	方程式+シミュレーション	46	18	49
E	最短路探索	52	33	40
F	計算幾何	6	3	130
G	アルゴリズム	31	5	67
H	幾何	50	9	80
I	動的計画法	24	6	60
J	正規表現, 探索	13	0	—

問題 A. Equal Sum Sets

入力として与えられた要素数・要素の値の上限・要素の和に対し、それらを満たすような正の整数からなる集合が何通りあるかを求める問題である。入力の値を小さく設定しているため複雑なロジックを組む必要はなく、簡単な問題である。最初の正解はコンテスト開始の 6 分後であり、多くのチームが早い時間に正解を出した。また、全てのチームが正解しただけでなく、誤答とエラーが合わせて 2 回と大変少なかった。

問題 B. The Last Ant

一本道のトンネル内で、蟻がすべて同じ速さで右向き、または、左向きに動いている。トンネル内で座標が整数となる場所は狭くなっており、そこで 2 匹の蟻が出会うと蟻は向きを反対方向へ変えて進み、それ以外の場所では蟻はすれ違うものとする。このとき、与えられた初期配置から始めて、最後にトンネルから出る蟻がどれかを答えさせる問題である。この問題も問題の大きさが小さく設定されており、単純なシミュレーションによって解を求めることができる。この問題も全チームが正解している。

問題 C. Count the Regions

xy 平面上にいくつかの長方形が与えられるとき、それらの長方形によって分けられる領域の数を求める問題である。長方形の各辺は x 軸または y 軸と平行である。長方形の数は少ないが、考慮すべき座標値の範囲は非常に大きい。そのため、範囲内の 1×1 の領域をすべて調べるような方法では間に合わない。長方形の辺として与えられる座標（とその前後）のみを調べるような方法をとる必要がある。31 チームがこの問題に正解している。一方で、誤答が 10 チーム、ランタイムエラーや制限時間オーバーが 6 チームと多かった。

問題 D. Clock Hands

時針・分針・秒針の 3 つの針を持つ H 時間制の時計がある。指定された時刻以降で、「針が互いに重ならない」「秒針とその他の 2 つの針の間の角度が等しい」という 2 つの条件を最初に満たす時刻を正確に求めるのが問題である。その時刻は分数の形で表されるため、その分母を（式変形による）計算で求める必要がある。分母が求まればプログラミングは比較的容易であるが、解答提出は 19 チームと難易度が中程度の問題としては少なかった。

問題 E. Dragon's Cruller

トラス上の 8 パズルを解く問題である。横方向と縦方向にピースをスライドするコストはそれぞれデータセットごとに与えられ、与えられた初期配置から目標配置を得る最小コストを求めなければならない。何らかのヒューリスティック関数を用いた A* アルゴリズムを用いると速いプログラムを書くことができるが、ダイクストラ法であっても制限時間内に解を求めることができるよう、問題の大きさを設定した。素直な問題であり、また、問題文に隣接マスの一覧を表で与えたことも影響してか、33 チームがこの問題に正解しており、A、B に続いて 3 番目に正解数が多い問題であった。

問題 F. Directional Resemblance

3 次元ベクトルが多数与えられる。それらのベクトルのうち、間の角度が最も小さくなるようなベクトルの組を求めるのが問題である。ただし、角度が 0 となるものは除外する。入力として与えられるベクトルの数が大変多いため、すべてのベクトルの組について計算するアルゴリズムでは制限時間を越えてしまう。最近点対問題（closest pair of points problem）を解くアルゴリズム（の考え方）を、球面上の点に対して応用することが求められる難しい問題であった。そのため、正解したのは 3 チームだけであり、問題 J を除いて最も正解チーム数が少ない結果となった。

問題 G. Longest Chain

3 次元空間内に多数の点が与えられる。それらの点から、x 座標・y 座標・z 座標がいずれも大きくなる点を選んでいくとき、最大でいくつの点を選べるかを答える問題である。この問題も、計算量が $O(n^2)$ となるような単純な方法では制限時間内に解を求めることができず、アルゴリズムを考える力が重要となる。審判団の想定解法は、与えられた点をどれかの座標についてソートした上で、k 個め ($k > 0$) の点になり得て他の点より小さいものから成る階段状のデータ構造を作り、これに対して二分探索を行うものである。12 チームから解答が提出された、正解は 5 チームのみであった。

なお、問題 F と問題 G では、入力のデータサイズを抑えるため、プログラム断片で与えられた方法で生成されるデータを併用する形式をとった。

問題 H. Don't Burst the Balloon

底面が正方形である箱と鉛直に立った針が与えられる。それらの針で割れることなく置くことができるような最大の風船の大きさを求めるのが問題である。この問題を解析的に解くのは、針に横から接する・針の上部に接する・箱に接するなど場合分けの条件が多く、かつ式変形も大変であるため非常に困難である。審判団の想定解法は、ある大きさの風船を置けるかを判定する関数を記述し、それを用いて風船の大きさについて二分探索を行うというものである。9 チームが正解し、3 次元幾何の問題としては正解が多かつ

た.

問題 I. Hidden Tree

二分木の各内部ノードから見てその左右の部分木の葉の重みの和が等しいとき、その二分木は均衡していると呼ぶ。与えられる数の列からいくつかの値を選択し、それらの数がある均衡した二分木の葉を並べたものとなるようにしたい。最大でいくつの数を選べるかを求めるのが問題である。制限時間内に解を求める動的計画法のアルゴリズムを導くには、(1) 均衡した木の葉には、 $a \times 2^k$ の形の値のみしか出現しないこと、(2) 均衡した木の一部を考える際に、木の形を覚えておく必要はなく、葉の数と葉の値の和だけ覚えておけば十分であること、に気付くことが必要である。難しい問題であるが、アルゴリズムが分かればプログラムは短く書けることもあり、開始の 60 分後に正解したチームもあった。

問題 J. C(O|W|A*RD*|S)* CROSSWORD Puzzle

正規表現をヒントとするクロスワードパズルを解く問題である。この問題では、言語に付随する正規表現ライブラリの使用は禁止した。与えられた正規表現を非決定性有限オートマトン (NFA) にし、さらに適切な順序で探索を行い、枝刈りをしっかり行うなど、ロジックの組み方の点でもプログラムコードの量の点でも難易度の非常に高い問題である。1 チームから解答プログラムの提出があったが、残念ながら正解には至らなかった。