HKOI Senior Q2 (Tournament) Editorial

Alex Tung
alex20030190@yahoo.com.hk

27 January 2018

Table of Contents

Task Description

- Single round-robin tournament with three teams (Alpha, Beta, Gamma).

Task Description

- Single round-robin tournament with three teams (Alpha, Beta, Gamma).
- In each match, two teams play against each other. Each team scores and concedes goals.

Task Description

- Single round-robin tournament with three teams (Alpha, Beta, Gamma).
- In each match, two teams play against each other. Each team scores and concedes goals.
- If (goals scored) $>$ (goals conceded), get W points.
- If (goals scored) $=$ (goals conceded), get D points.
- If (goals scored) $<$ (goals conceded), get 0 points.

Task Description

- Single round-robin tournament with three teams (Alpha, Beta, Gamma).
- In each match, two teams play against each other. Each team scores and concedes goals.
- If (goals scored) $>$ (goals conceded), get W points.
- If (goals scored) $=$ (goals conceded), get D points.
- If (goals scored) $<$ (goals conceded), get 0 points.
- For each team, you know its total #points, total #goals scored, and total #goals conceded.

Task Description

- Single round-robin tournament with three teams (Alpha, Beta, Gamma).
- In each match, two teams play against each other. Each team scores and concedes goals.
- If (goals scored) $>$ (goals conceded), get W points.
- If (goals scored) $=$ (goals conceded), get D points.
- If (goals scored) $<$ (goals conceded), get 0 points.
- For each team, you know its total #points, total #goals scored, and total #goals conceded.
- Output a list of match results, which matches the given info.

Sample IO

Sample Input 1

```
3 1
6 2 0
1 0 1
1 0 1
```

Sample Output 1

```
Alpha 1 - 0 Beta
Alpha 1 - 0 Gamma
Beta 0 - 0 Gamma
```

Sample Input 2

```
3 1
6 3 0
1 0 1
1 0 1
```

Sample Output 2

```
Impossible
```

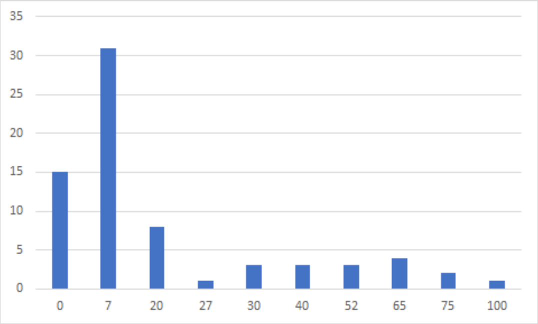
Constraints

For all cases:
$0 \le D \le W \le 5$
$0 \le P_A, P_B, P_G \le 2W$
$0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 10^9$

	Points	Constraints
1	7	$S_A = S_B = S_G = C_A = C_B = C_G = 0$
2	13	$W > 2D$ $P_A = P_B = P_G = 2D$ $S_A = C_A$ $S_B = C_B$ $S_G = C_G$
3	20	$0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 20$
4	25	$0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 10^6$
5	10	$W = 0$
6	25	No additional constraints

Statistics

Attempts: 71
Mean: 18.028
Stddev: 22.72
Top scores: 100 (hkoi201516-28, 1:22), 75 (s14318, hkoi201516-27), 65 (4 contestants)
Score distribution:



Comments

- An ad-hoc, hard problem.

Comments

- An ad-hoc, hard problem.
- Requires (mathematical?) insight + good coding skills.

Comments

- An ad-hoc, hard problem.
- Requires (mathematical?) insight $+$ good coding skills.
- I am glad that Max $= 100$:)

Comments

- An ad-hoc, hard problem.
- Requires (mathematical?) insight $+$ good coding skills.
- I am glad that Max $= 100$:)
- It has really easy subtasks.

Subtask 1

Subtask 1 (7 points): No goals scored, no goals conceded

Subtask 1

Subtask 1 (7 points): No goals scored, no goals conceded

- Each match should end in a draw.

Subtask 1

Subtask 1 (7 points): No goals scored, no goals conceded

- Each match should end in a draw.
- Suffices to check whether $P_A = P_B = P_G = 2 \times D$.

Subtask 1

Subtask 1 (7 points): No goals scored, no goals conceded

- Each match should end in a draw.
- Suffices to check whether $P_A = P_B = P_G = 2 \times D$.
- If true, just output

```
Alpha 0 - 0 Beta
Beta 0 - 0 Gamma
Gamma 0 - 0 Alpha
```

  Otherwise, output `Impossible`.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.
- Let AB be #goals scored by team Alpha in the Alpha vs. Beta match.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.
- Let AB be #goals scored by team Alpha in the Alpha vs. Beta match.
- Let BA be #goals scored by team Beta in the Alpha vs. Beta match.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.
- Let AB be #goals scored by team Alpha in the Alpha vs. Beta match.
- Let BA be #goals scored by team Beta in the Alpha vs. Beta match.
- Similar for AG, GA, BG, and GB.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.
- Let AB be #goals scored by team Alpha in the Alpha vs. Beta match.
- Let BA be #goals scored by team Beta in the Alpha vs. Beta match.
- Similar for AG, GA, BG, and GB.
- For this subtask, $AB = BA$, $AG = GA$, $BG = GB$.

Subtask 2

Subtask 2 (13 points):
$W > 2D, P_A = P_B = P_G = 2D, S_A = C_A, S_B = C_B, S_G = C_G$

- Looks terrible, but all it says is that (if a solution exists) all matches should be draws. Otherwise, some team (say Alpha) wins a match and so $P_A \geq W > 2D = P_A$, contradiction.
- Let AB be #goals scored by team Alpha in the Alpha vs. Beta match.
- Let BA be #goals scored by team Beta in the Alpha vs. Beta match.
- Similar for AG, GA, BG, and GB.
- For this subtask, $AB = BA$, $AG = GA$, $BG = GB$.
- Then we are solving the following system of equations:

$$\begin{cases} AB + AG & = & S_A \\ AB + BG & = & S_B \\ AG + BG & = & S_G \end{cases}$$

Subtask 2

- We are solving the following system of equations:

$$\begin{cases} AB + AG &= S_A \\ AB + BG &= S_B \\ AG + BG &= S_G \end{cases}$$

Subtask 2

- We are solving the following system of equations:

$$\begin{cases} AB + AG & = & S_A \\ AB + BG & = & S_B \\ AG + BG & = & S_G \end{cases}$$

- Add up the equations to get $2 \times (AB + AG + BG) = S_A + S_B + S_G$.

Subtask 2

- We are solving the following system of equations:

$$\begin{cases} AB + AG &= S_A \\ AB + BG &= S_B \\ AG + BG &= S_G \end{cases}$$

- Add up the equations to get $2 \times (AB + AG + BG) = S_A + S_B + S_G$.
- Divide by two, then we know the value of $AB + AG + BG$.

Subtask 2

- We are solving the following system of equations:

$$\begin{cases} AB + AG &= S_A \\ AB + BG &= S_B \\ AG + BG &= S_G \end{cases}$$

- Add up the equations to get $2 \times (AB + AG + BG) = S_A + S_B + S_G$.
- Divide by two, then we know the value of $AB + AG + BG$.
- If the value is not integer, output `Impossible`.

Subtask 2

- We are solving the following system of equations:

$$\begin{cases} AB + AG &= S_A \\ AB + BG &= S_B \\ AG + BG &= S_G \end{cases}$$

- Add up the equations to get $2 \times (AB + AG + BG) = S_A + S_B + S_G$.
- Divide by two, then we know the value of $AB + AG + BG$.
- If the value is not integer, output `Impossible`.
- Otherwise, subtract the equations above to get AB, AG, BG. (Check that they are non-negative!)

Subtask 3

Subtask 3 (20 points): $0 \leq S_A, S_B, S_G, C_A, C_B, C_G \leq 20$

Subtask 3

Subtask 3 (20 points): $0 \leq S_A, S_B, S_G, C_A, C_B, C_G \leq 20$

- Just write six layers of for-loops to exhaust all match results :)

Subtask 3

Subtask 3 (20 points): $0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 20$

- Just write six layers of for-loops to exhaust all match results :)
- There are at most 21^6 combinations to check.

Subtask 3

Subtask 3 (20 points): $0 \leq S_A, S_B, S_G, C_A, C_B, C_G \leq 20$

- Just write six layers of for-loops to exhaust all match results :)
- There are at most 21^6 combinations to check.
- Time complexity: $O(R^6)$, where R is the input range.

Subtask 4

Subtask 4 (25 points): $0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 10^6$

Claim

If we fix any of the six scores, we may deduce the remaining five.

Proof

This is because we are solving the following system of equations:

$$\begin{cases} AB + AG & = & S_A \\ AG + BG & = & C_G \\ BG + BA & = & S_B \\ BA + GA & = & C_A \\ GA + GB & = & S_G \\ GB + AB & = & C_B \end{cases}$$

Subtask 4

Subtask 4 (25 points): $0 \le S_A, S_B, S_G, C_A, C_B, C_G \le 10^6$

Claim

If we fix any of the six scores, we may deduce the remaining five.

Proof

This is because we are solving the following system of equations:

$$\begin{cases} AB + AG &= S_A \\ AG + BG &= C_G \\ BG + BA &= S_B \\ BA + GA &= C_A \\ GA + GB &= S_G \\ GB + AB &= C_B \end{cases}$$

- This improves the algorithm to $O(R)$.

Subtask 5

Subtask 5 (10 points): $W = 0$

Subtask 5

Subtask 5 (10 points): $W = 0$

- For this subtask, we don't need to care about points.

Subtask 5

Subtask 5 (10 points): $W = 0$

- For this subtask, we don't need to care about points.
- Say we try $AB = x$. Then, we have

$$\begin{cases} AG &= S_A - AB &= S_A - x \\ BG &= C_G - AG &= C_G - S_A + x \\ BA &= S_B - BG &= S_B - C_G + S_A - x \\ GA &= C_A - BA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - GA &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

Subtask 5

Subtask 5 (10 points): $W = 0$

- For this subtask, we don't need to care about points.
- Say we try $AB = x$. Then, we have

$$\begin{cases} AG & = S_A - AB & = S_A - x \\ BG & = C_G - AG & = C_G - S_A + x \\ BA & = S_B - BG & = S_B - C_G + S_A - x \\ GA & = C_A - BA & = C_A - S_B + C_G - S_A + x \\ GB & = S_G - GA & = S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- Of course we should have $S_A + S_B + S_G = C_A + C_B + C_G$.

Subtask 5

Subtask 5 (10 points): $W = 0$

- For this subtask, we don't need to care about points.
- Say we try $AB = x$. Then, we have

$$\begin{cases} AG & = S_A - AB & = S_A - x \\ BG & = C_G - AG & = C_G - S_A + x \\ BA & = S_B - BG & = S_B - C_G + S_A - x \\ GA & = C_A - BA & = C_A - S_B + C_G - S_A + x \\ GB & = S_G - GA & = S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- Of course we should have $S_A + S_B + S_G = C_A + C_B + C_G$.
- Then, $AB = x$ is valid iff AG, BG, BA, GB, GA are non-negative.

Subtask 5

Subtask 5 (10 points): $W = 0$

- For this subtask, we don't need to care about points.
- Say we try $AB = x$. Then, we have

$$\begin{cases} AG &= S_A - AB &= S_A - x \\ BG &= C_G - AG &= C_G - S_A + x \\ BA &= S_B - BG &= S_B - C_G + S_A - x \\ GA &= C_A - BA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - GA &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- Of course we should have $S_A + S_B + S_G = C_A + C_B + C_G$.
- Then, $AB = x$ is valid iff AG, BG, BA, GB, GA are non-negative.
- The above equations give the valid range of x.

Subtask 6

Subtask 6 (25 points): No additional constraints

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.
- We know how to express other variables in terms of x and constants.

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.
- We know how to express other variables in terms of x and constants.
- Now, exhaust the match results. Just win/draw/lose, no scores.

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.
- We know how to express other variables in terms of x and constants.
- Now, exhaust the match results. Just win/draw/lose, no scores.
- So there are $3^3 = 27$ possibilities.

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.
- We know how to express other variables in terms of x and constants.
- Now, exhaust the match results. Just win/draw/lose, no scores.
- So there are $3^3 = 27$ possibilities.
- Check if points ($P_A, P_B,$ and P_G) match, then try to find a valid x.

Subtask 6

Subtask 6 (25 points): No additional constraints

- Again, let $AB = x$.
- We know how to express other variables in terms of x and constants.
- Now, exhaust the match results. Just win/draw/lose, no scores.
- So there are $3^3 = 27$ possibilities.
- Check if points ($P_A, P_B,$ and P_G) match, then try to find a valid x.
- This can be done, again, by solving inequalities (and equalities).

Subtask 6

- Recall:

$$\begin{cases} AG &= S_A - x \\ BG &= C_G - S_A + x \\ BA &= S_B - C_G + S_A - x \\ GA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

Subtask 6

- Recall:

$$\begin{cases} AG &= S_A - x \\ BG &= C_G - S_A + x \\ BA &= S_B - C_G + S_A - x \\ GA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- For example, requiring Alpha defeats Beta means that $AB > BA$, or $x > S_B - C_G + S_A - x$.

Subtask 6

- Recall:

$$\begin{cases} AG &= S_A - x \\ BG &= C_G - S_A + x \\ BA &= S_B - C_G + S_A - x \\ GA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- For example, requiring Alpha defeats Beta means that $AB > BA$, or $x > S_B - C_G + S_A - x$.
- Requiring Beta draws against Gamma means that $BG = GB$, or $C_G - S_A + x = S_G - C_A + S_B - C_G + S_A - x$.

Subtask 6

- Recall:

$$\begin{cases} AG &= S_A - x \\ BG &= C_G - S_A + x \\ BA &= S_B - C_G + S_A - x \\ GA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- For example, requiring Alpha defeats Beta means that $AB > BA$, or $x > S_B - C_G + S_A - x$.
- Requiring Beta draws against Gamma means that $BG = GB$, or $C_G - S_A + x = S_G - C_A + S_B - C_G + S_A - x$.
- These constraints give the valid range of x.

Subtask 6

- Recall:

$$\begin{cases} AG &= S_A - x \\ BG &= C_G - S_A + x \\ BA &= S_B - C_G + S_A - x \\ GA &= C_A - S_B + C_G - S_A + x \\ GB &= S_G - C_A + S_B - C_G + S_A - x \end{cases}$$

- For example, requiring Alpha defeats Beta means that $AB > BA$, or $x > S_B - C_G + S_A - x$.
- Requiring Beta draws against Gamma means that $BG = GB$, or $C_G - S_A + x = S_G - C_A + S_B - C_G + S_A - x$.
- These constraints give the valid range of x.
- Alternatively, one may use binary search to find one possible x.

Subtask 6

- But a simpler solution exists!

Subtask 6

- But a simpler solution exists!
- Exhausting the match results (win/draw/lose) means that we require $AB - BA$, $BG - GB$, $GA - AG$ to be positive/zero/negative.

Subtask 6

- But a simpler solution exists!
- Exhausting the match results (win/draw/lose) means that we require $AB - BA$, $BG - GB$, $GA - AG$ to be positive/zero/negative.

$$\begin{cases} AB - BA &= 2x - f(A,B) \\ BG - GB &= 2x - f(B,G) \\ GA - AG &= 2x - f(G,A) \end{cases}$$

Subtask 6

- But a simpler solution exists!
- Exhausting the match results (win/draw/lose) means that we require $AB - BA$, $BG - GB$, $GA - AG$ to be positive/zero/negative.

$$\begin{cases} AB - BA & = 2x - f(A,B) \\ BG - GB & = 2x - f(B,G) \\ GA - AG & = 2x - f(G,A) \end{cases}$$

- $f(A,B), f(B,G), f(G,A)$ are constants.

Subtask 6

- But a simpler solution exists!
- Exhausting the match results (win/draw/lose) means that we require $AB - BA$, $BG - GB$, $GA - AG$ to be positive/zero/negative.

$$\begin{cases} AB - BA & = 2x - f(A, B) \\ BG - GB & = 2x - f(B, G) \\ GA - AG & = 2x - f(G, A) \end{cases}$$

- $f(A, B), f(B, G), f(G, A)$ are constants.
- They may be computed using the equations in the previous slide.

Subtask 6

- But a simpler solution exists!
- Exhausting the match results (win/draw/lose) means that we require $AB - BA$, $BG - GB$, $GA - AG$ to be positive/zero/negative.

$$\begin{cases} AB - BA &= 2x - f(A, B) \\ BG - GB &= 2x - f(B, G) \\ GA - AG &= 2x - f(G, A) \end{cases}$$

- $f(A, B), f(B, G), f(G, A)$ are constants.
- They may be computed using the equations in the previous slide.
- Therefore, it suffices to check all x *"near"* $\frac{f(A,B)}{2}$, $\frac{f(B,G)}{2}$, or $\frac{f(G,A)}{2}$.

Happy Ending? Not Yet!

- Checking these values of x is **not enough**!

Happy Ending? Not Yet!

- Checking these values of x is **not enough**!

- One also needs to check all x such that either of AB, BA, AG, GA, BG, GB is zero.

Happy Ending? Not Yet!

- Checking these values of x is **not enough**!

2017-12-25 13:01:27	kctung - RB教教徒	S182 - Tournament
Wrong Answer	Score: 95.821	

- One also needs to check all x such that either of AB, BA, AG, GA, BG, GB is zero.
- Like this:

```
test((gs[1] - gc[2] + gs[0]) / 2);
test((gc[1] - gs[2] + gs[0]) / 2);
test((gc[1] - gc[2] + gs[0]) / 2);
test(0);
test(gs[0]);
test(gs[0] - gc[2]);
test(gs[0] - gc[2] + gs[1]);
test(gs[0] - gc[2] + gs[1] - gc[0]);
test(gs[0] - gc[2] + gs[1] - gc[0] + gs[2]);
```

The End

- Questions?