Algorithms without Programming

Marcin Kubica, **Jakub Radoszewski**

Poland

IOI Conference August 16, 2010

Plan of the Presentation

1. Introduction and discussion of previous work
2. Examples
3. Conclusions

The Objectives

1. Programming contests popularize computer science.

The Objectives

1. Programming contests popularize computer science.
2. Programming might be too much for the first step.

The Objectives

1. Programming contests popularize computer science.
2. Programming might be too much for the first step.
3. We can first introduce algorithms, and programming afterwards.

The Objectives

1. Programming contests popularize computer science.
2. Programming might be too much for the first step.
3. We can first introduce algorithms, and programming afterwards.
4. Problems formulated as (mathematical) puzzles, solutions involve algorithmics.

Previous Work

- Bebras competition (`www.bebras.org`; Dagiene & Futschek, ISSEP 2008; Dagiene, ISSEP 2010)
- South African olympiad (Merry et al., *Olymp. in Inf.* 2, 2008)
- Australian Informatics Competition (Clark, *The Austr. Math. Teach.* 62, 2006)
- Ugale competition (Opmanis, *Olymp. in Inf.* 3, 2009)
- Project Euler (`projecteuler.net`)
- Internet Problem Solving Contest (`ipsc.ksp.sk`)

Common Ideas

- Attractively-formulated problems involving computer science (in the statement, in the solution or in both).
- Solutions require creativity rather than prior knowledge.

Bebras Competition

Bebras Competition

- problems should be solvable by hand within 3 minutes each, forming a competition round,
- one of the main goals is popularization of computer science.

Our Ideas

- puzzle-type problems which could be harder and contain hints, solvable using a pen and a piece of paper,
- one of the main goals is popularization of computer science *among pupils interested in mathematics*.

Ugale Competition, IPSC

Ugale & IPSC Competitions

- introducing new and attractive types of computer science problems.

Our Ideas

- any computer science is hidden within the ostensibly purely mathematical solution.

Project Euler

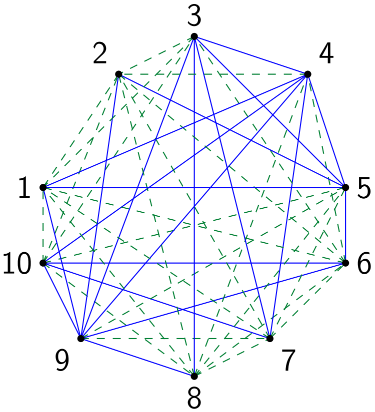
Project Euler

- problems with purely mathematical formulation, but solutions employing computing,
- mostly devoted to number-theoretic problems.

Our Ideas

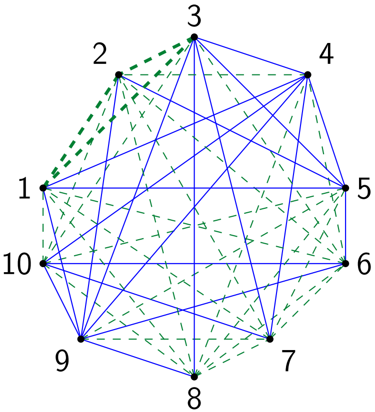
- problems with purely mathematical formulation, solutions employing well-hidden methods of computer science.

Example 1: Uni-color Triangles



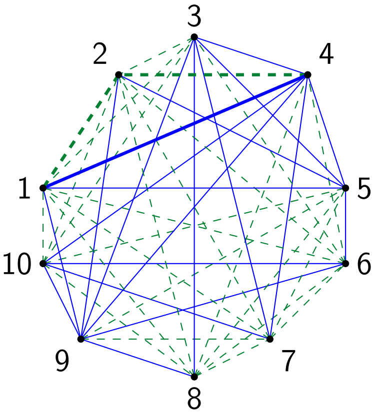
How many uni-color triangles, with vertices at the given points, are present in the figure?

Straightforward Approach



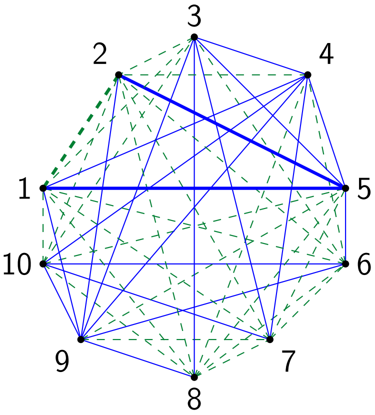
(1,2,3) forms a uni-color triangle.

Straightforward Approach



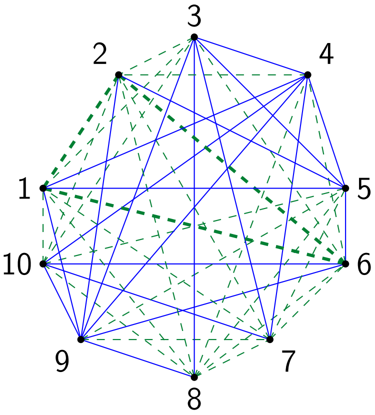
(1,2,4) is not a uni-color triangle.

Straightforward Approach



(1,2,5) is not a uni-color triangle.

Straightforward Approach



(1,2,6) forms a uni-color triangle...

A Different Approach

The total number of triangles is:

$$\binom{10}{3} = \frac{10 \cdot 9 \cdot 8}{1 \cdot 2 \cdot 3} = 120.$$

We need a more efficient approach!

A Different Approach

The total number of triangles is:

$$\binom{10}{3} = \frac{10 \cdot 9 \cdot 8}{1 \cdot 2 \cdot 3} = 120.$$

We need a more efficient approach!

Hint: count multi-color triangles.

Multi-color Triangles

1. Each multi-color triangle contains 2 vertices in which 2 sides of different colors meet.

Multi-color Triangles

1. Each multi-color triangle contains 2 vertices in which 2 sides of different colors meet.
2. Every pair of lines of different colors meeting at a point induces one multi-color triangle.

Multi-color Triangles

1. Each multi-color triangle contains 2 vertices in which 2 sides of different colors meet.
2. Every pair of lines of different colors meeting at a point induces one multi-color triangle.
3. By considering all such pairs of lines, we will count twice the number of multi-color triangles.

Multi-color Triangles

	1	2	3	4	5	6	7	8	9	10
1		g	g	b	b	g	g	g	b	g
2	g		g	g	b	g	g	g	b	g
3	g	g		b	b	g	b	b	b	g
4	b	g	b		b	g	b	g	b	b
5	b	b	b	b		b	g	g	g	g
6	g	g	g	g	b		g	g	b	b
7	g	g	b	b	g	g		g	g	b
8	g	g	b	g	g	g	g		b	g
9	b	b	b	b	g	b	g	b		b
10	g	g	g	b	g	b	b	g	b	

Multi-color Triangles

	1	2	3	4	5	6	7	8	9	10	(#b, #g)
1		g	g	b	b	g	g	g	b	g	**(3, 6)**
2	g		g	g	b	g	g	g	b	g	**(2, 7)**
3	g	g		b	b	g	b	b	b	g	**(5, 4)**
4	b	g	b		b	g	b	g	b	b	**(6, 3)**
5	b	b	b	b		b	g	g	g	g	**(5, 4)**
6	g	g	g	g	b		g	g	b	b	**(3, 6)**
7	g	g	b	b	g	g		g	g	b	**(3, 6)**
8	g	g	b	g	g	g	g		b	g	**(2, 7)**
9	b	b	b	b	g	b	g	b		b	**(7, 2)**
10	g	g	g	b	g	b	b	g	b		**(4, 5)**

Multi-color Triangles

	1	2	3	4	5	6	7	8	9	10	(#b, #g)	#b·#g
1		g	g	b	b	g	g	g	b	g	(3, 6)	**18**
2	g		g	g	b	g	g	g	b	g	(2, 7)	**14**
3	g	g		b	b	g	b	b	b	g	(5, 4)	**20**
4	b	g	b		b	g	b	g	b	b	(6, 3)	**18**
5	b	b	b	b		b	g	g	g	g	(5, 4)	**20**
6	g	g	g	g	b		g	g	b	b	(3, 6)	**18**
7	g	g	b	b	g	g		g	g	b	(3, 6)	**18**
8	g	g	b	g	g	g	g		b	g	(2, 7)	**14**
9	b	b	b	b	g	b	g	b		b	(7, 2)	**14**
10	g	g	g	b	g	b	b	g	b		(4, 5)	**20**

The Solution

In total, the number of multi-color triangles:

$$18+14+20+18+20+18+18+14+14+20 = 174.$$

The Solution

In total, the number of multi-color triangles:

$$18+14+20+18+20+18+18+14+14+20 = 174.$$

The number of uni-color triangles:

$$120 - \frac{174}{2} = 33.$$

Methodological Comments

- Brute force solution has $\Theta(n^3)$ time complexity.
- The large number of possible triangles (120) strongly encourages to seek for a better solution.
- The improved solution has $\Theta(n^2)$ time complexity and is easy to perform by hand.
- The hint makes the task a lot easier.

Example 2: Coins

We are given 11 coins of the following values:

$$7,\ 300,\ 35,\ 83,\ 1,\ 17,\ 2,\ 1,\ 17,\ 170,\ 5.$$

What is the smallest (positive integer) amount of money that cannot be paid using the coins?

E.g., $59 = 35 + 17 + 5 + 1 + 1$.

On the other hand, the sum

$$639 = (7 + 300 + 35 + 83 + 1 + 17 + 2 + 1 + 17 + 170 + 5) + 1$$

cannot be paid.

First Observations

Recall the values of coins:

$$7,\ 300,\ 35,\ 83,\ 1,\ 17,\ 2,\ 1,\ 17,\ 170,\ 5$$

$1 = 1,\ \ 2 = 2,\ \ 3 = 2 + 1,\ \ 4 = 2 + 1 + 1,\ \ 5 = 5,\ \ 6 = 5 + 1,$
$7 = 5 + 1 + 1,\ \ 8 = 5 + 2 + 1, \ldots$

First Observations

Recall the values of coins:

$$7,\ 300,\ 35,\ 83,\ 1,\ 17,\ 2,\ 1,\ 17,\ 170,\ 5$$

$1 = 1,\ \ 2 = 2,\ \ 3 = 2 + 1,\ \ 4 = 2 + 1 + 1,\ \ 5 = 5,\ \ 6 = 5 + 1,$
$7 = 5 + 1 + 1,\ \ 8 = 5 + 2 + 1,\ \ldots$

Clearly, we need something smarter. Let us order the values of the coins:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

First Observations

Recall the values of coins:

$$7,\ 300,\ 35,\ 83,\ 1,\ 17,\ 2,\ 1,\ 17,\ 170,\ 5$$

$1 = 1,\ \ 2 = 2,\ \ 3 = 2 + 1,\ \ 4 = 2 + 1 + 1,\ \ 5 = 5,\ \ 6 = 5 + 1,$
$7 = 5 + 1 + 1,\ \ 8 = 5 + 2 + 1,\ \ldots$

Clearly, we need something smarter. Let us order the values of the coins:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

For each prefix of the list, let us find out what amounts of money can be paid with the corresponding coins.

Towards the Solution

Recall the ordered list:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

Towards the Solution

Recall the ordered list:

$$1,\ 1,\ 2,\ 5,\ 7,\ 17,\ 17,\ 35,\ 83,\ 170,\ 300.$$

prefix 1: amounts $[0, 1]$

Towards the Solution

Recall the ordered list:

$$1,\ 1,\ 2,\ 5,\ 7,\ 17,\ 17,\ 35,\ 83,\ 170,\ 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$

Towards the Solution

Recall the ordered list:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$

Towards the Solution

Recall the ordered list:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$
prefix 1, 1, 2, 5: amounts $[0, 9]$

Towards the Solution

Recall the ordered list:

$$1,\ 1,\ 2,\ 5,\ 7,\ 17,\ 17,\ 35,\ 83,\ 170,\ 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$
prefix 1, 1, 2, 5: amounts $[0, 9]$
prefix 1, 1, 2, 5, 7: amounts $[0, 16]$

Towards the Solution

Recall the ordered list:

$$1, 1, 2, 5, 7, 17, 17, 35, 83, 170, 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$
prefix 1, 1, 2, 5: amounts $[0, 9]$
prefix 1, 1, 2, 5, 7: amounts $[0, 16]$
prefix 1, 1, 2, 5, 7, 17: amounts $[0, 33]$

Towards the Solution

Recall the ordered list:

$$1,\ 1,\ 2,\ 5,\ 7,\ 17,\ 17,\ 35,\ 83,\ 170,\ 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$
prefix 1, 1, 2, 5: amounts $[0, 9]$
prefix 1, 1, 2, 5, 7: amounts $[0, 16]$
prefix 1, 1, 2, 5, 7, 17: amounts $[0, 33]$

Continue until the prefix 1, 1, 2, 5, 7, 17, 17, 35, 83, for which the amounts are in $[0, 168]$, but $170 > 168 + 1$.

Towards the Solution

Recall the ordered list:

$$1,\ 1,\ 2,\ 5,\ 7,\ 17,\ 17,\ 35,\ 83,\ 170,\ 300.$$

prefix 1: amounts $[0, 1]$
prefix 1, 1: amounts $[0, 2]$
prefix 1, 1, 2: amounts $[0, 4]$
prefix 1, 1, 2, 5: amounts $[0, 9]$
prefix 1, 1, 2, 5, 7: amounts $[0, 16]$
prefix 1, 1, 2, 5, 7, 17: amounts $[0, 33]$

Continue until the prefix 1, 1, 2, 5, 7, 17, 17, 35, 83, for which the amounts are in $[0, 168]$, but $170 > 168 + 1$.

Thus, the answer is **169**.

Methodological Comments

- Brute force solution has $\Theta(n \cdot S)$ time complexity. Either dynamic programming or "guessing the decomposition".
- Optimal solution has $O(n)$ time complexity. It is in a way greedy.

Example 3: Divisible by 13?

Does the following sequence of numbers:

$$(1, 1, 9, 7, 12, 4, 12, 5, 8, 2, 7, 2, 10, 2, 3)$$

contain a non-empty, continuous subsequence, whose sum is divisible by 13? If so, what is the number of such subsequences?

Example 4: Palindromic numbers

A number is called *palindromic* if its decimal number is a palindrome, e.g., 5, 22 and 21 312.

How many palindromic numbers are there in the interval $[285\,924, 84\,633\,902]$?

Conclusions

Each of the presented tasks was a small instance of a regular programming contest task.

Is every such task good for our purposes?

Algorithmic Puzzles

What should we take into consideration?

- The model solution minimizes thinking-time + "execution"-time.

Algorithmic Puzzles

What should we take into consideration?

- The model solution minimizes thinking-time + "execution"-time.
- It is better to exclude problems which require classical algorithms and advanced techniques.

Algorithmic Puzzles

What should we take into consideration?

- The model solution minimizes thinking-time + "execution"-time.
- It is better to exclude problems which require classical algorithms and advanced techniques.
- It is good to have a simple but slow solution.

Algorithmic Puzzles

What should we take into consideration?

- The model solution minimizes thinking-time + "execution"-time.
- It is better to exclude problems which require classical algorithms and advanced techniques.
- It is good to have a simple but slow solution.
- The hints may serve as "after-teasers".

The End

Thank you for your attention!