Programming Competitions for ENS Lyon Training Camp: Problem solving tips and tricks

Vitaly Aksenov, Fedor Tsarev

October 27, 2015

Start of the contest

- Write a template
- Read problem statements:
  - Split the problems between teammates.
  - If you see the short problem, you need to read it first. (short problems are easiest most of the time)
- Do not try to solve the first problem you have read, find the **easiest** problem
- Each member must know his role at the start of the contest

Problem status

№	Member 1	Member 2	Member 3	Comments
A	+			Graph theory
B	+			
C			+	
D			+	
…				

One hour from the start of the contest

№	Member 1	Member 2	Member 3	Comments
A	+			Graph theory
B	+	+	+	KMP
C	+	+	+	DP over subsets
D	+	+	+	Geometry
...	+	+	+	

Easiest problem - first!

- Time needed to solve each of the problems:
  $t_1, t_2, \ldots, t_n$
- Penalty time:
  $$\sum_{i=1}^{n} \sum_{j=1}^{i} t_i = n \cdot t_1 + (n-1) \cdot t_2 + \ldots + \cdot t_n$$

How to solve a problem

1. Read the statement carefully
2. Invent an algorithm
3. Ask your teammate to read the statement
4. Discuss the algorithm with another member of your team
5. Discuss the implementation with another member of your team
6. Write a program
7. While you are writing, ask your teammate to create testcases
8. Test your solution

Submitting a Solution

- Save the source file before submitting
- Compile it
- **Always** test the solution on example tests and on all tests you have:
  - Corner cases
  - Maximal tests (checking time limit and memory limit)
- Print the source code after submitting (on the official contest)

Getting Wrong Answer

- Do not panic!
- Print your code and read from it
- **Never** read the code from the screen! Somebody could work.
- If you find a bug do not fix it right now on the computer: there are could be more bugs in the program
- **Check for integer overflow**
- Fix the bug in the printout
- Try to explain your code to another member of your team
- Write a slower solution and stress-test with it

Debugging

- Debugging is a very time-consuming process
- You can't write code while debugging

Avoid using debugger!
Read the printout instead

Programming in pairs

- If the implementation is too difficult for you…
- If you are tired…
- If you have nothing to do…
- Try programming in pairs:
  - One of the team members writes the code
  - Another one is looking for mistakes

Last hour of the contest

- Your goal is to get "Accepted" for all problems you have started
- Do not start new problems if you have "old" ones
- If you do not have enough time for all problems choose some of them