ProgTeam Finals Week

Guide to Competing

Competition Structure

- Usually 6-14 problems
- Problems usually aren't sorted by difficulty!
- ICPC scoring = # of problems solved
  - Break ties with penalty
  - Penalty = Sum of submission times + 20 minutes per wrong submission
- Example:
  - Solved problem A at 15 minutes with 3 wrong submissions
  - Solved problem B at 45 minutes with 0 wrong submissions
  - Submitted 4 wrong submissions to problem C; Did not solve
  - Score = 2 Correct problems
  - Penalty = 15 + 45 + 3 * 20 = 120

Competition Advice

- Read input from stdin
  - Examples given in github
- Watch the scoreboard!
  - If a problem is being solved by many teams, it is probably easier than others
- Don’t get tunnel vision!
  - If you get stuck on a problem, rule of thumb is solve something else after 20 minutes of no progress
- Common pitfalls:
  - Is input format being handled correctly?
  - Is there any overflow? Is memory allocated correctly?
  - What are the edge cases? (N = 1? N = {max value}?)

ICPC-Specific Advice: How to spend your time

- First 5-15 minutes: read the problems!
  - Keep mental notes of which ones seem approachable
  - If there's one that seems easy, go ahead and try it (don't go too fast though!)
  - Watch the scoreboard! Easy problems get solved first
- First 2 hours
  - You'll probably solve most of the problems you're going to in this period
  - These are all the ones you more-or-less know how to do but need to figure out a few smaller details
- Last 3 hours
  - This is where you can tackle some of the trickier problems, and fix any bugs from the 2-hour period

Advanced Tips

- Pre-written templates are usually allowed
  - KACTL has a great library for C++
    - https://github.com/kth-competitive-programming/kactl/tree/main/content
- Occasionally there will be a problem where time limit is too small for Python/Java
  - Rule of thumb: 10 million operations is “safe” for Python. 50 million operations is “safe” for Java. 100 million operations is “safe” for C++.
  - Slower solutions *may* pass
    - Operations like modulo, division, memory allocation are much slower than operations like adding, subtracting, checking array index
- Read all the problems! Occasionally an easy problem gets missed by most teams.
- Always try to solve easier problems first
  - 5->20->65 is much lower penalty than 45->60->65