

# Introduction to ACM-ICPC and CCPC

The University of Hong Kong  
Department of Computer Science  
SUN Bintao

# Outline

- Introduction to ACM-ICPC and CCPC
- Upcoming contests
- Practice and training

# Introduction to ACM-ICPC and CCPC

# What are they?

- ACM International Collegiate Programming Contest.
- Official site: <https://icpc.baylor.edu/>
- The Association for Computing Machinery (ACM) is an international learned society for computing. It is the world's largest scientific and educational computing society.

# What are they?

- China Collegiate Programming Contest.
- Official site: <https://ccpc.io/>
- The 1<sup>st</sup> CCPC was held in 2015.

# Basic rules

- Solve about 10 programming tasks in 5 hours.
- 3 members per team.
- 1 computer.
- C++ or Java.

# Question style

- Problems are mathematically well-defined, with data range and input/output specifications.
- Example: <http://poj.org/problem?id=1001>
- No GUI, network etc. are needed. Actually they are not allowed.
- For C++, all you need is iostream and STL.

# How is it judged?

- Only source code is submitted and judged.
- Data-based judging. Your source code is compiled and run by the judge. The judge uses a preset dataset as input, and compare your output against the referenced answer.



# How is it judged?

- Your solution is judged as correct (**AC - Accepted**) only if
- It compiles successfully. Otherwise **CE - Compile Error**.
- It does not crash. Otherwise **RE - Runtime Error**.
- It uses limited amount of memory. Otherwise **MLE - Memory Limit Exceeded**.
- It terminates within the time limit. Otherwise **TLE - Time Limit Exceeded**.
- It outputs all the correct answers. Otherwise **WA - Wrong Answer**.

# An example

- <http://poj.org/status>

# How are teams ranked?

- Teams are firstly ranked according to the number of problems solved, then the total time used.
- The **total time** is the sum of the time consumed for each problem solved. The time consumed for a solved problem is the **time elapsed** from the beginning of the contest to the submittal of the first accepted run plus **20 penalty minutes** for every previously rejected run (Compile Error excluded) for that problem.

# Example

- Suppose you have the following submissions:
  - The 12th minute, problem A, correct;
  - The 34th minute, problem B, incorrect;
  - The 56th minute, problem C, incorrect;
  - The 78th minute, problem B, correct;
  - The 90th minute, problem A, incorrect.
- Then your time consumed is 12 for problem A,  $78 + 20 * 1 = 98$  for problem B and 0 for problem C. Your total time is thus 110.
- Contestants can check ranks **in real time**.
- Example: <https://icpc.baylor.edu/scoreboard/>

# A picture



So far, all accepted solutions for problem E use the easier but slower implementation that runs in  $n \log n$  time, said Michal Forisek. Follow

# Contest structure of ICPC

- University selection
  - We will select 3 teams this year.
- Asia regional contests (East Continent sub-region)
  - Our main battle field.
- World finals
  - Top teams in regional contests advance to world finals. About 120 teams in total around the world.

# Skills required

- Coding. Implement whatever algorithm you have in mind.
- Algorithm design. Knowledge of classic algorithms and algorithm design techniques. For example, Dijkstra's algorithm and dynamic programming.
- Programming and debugging onsite under pressure.

Upcoming contests



# Contest plan

- We plan to select 3 teams.
- ICPC rules: each team can go to at most 2 regional contests (weekends in Oct and Nov) + East Continent Final (Dec 14-15).
- Usually, a silver medal or above at any regional site can secure a ticket to EC Final. Sometimes a bronze medal is also enough.
- CCPC (Final): Nov 16-17.

# Contest plan

- Depending on budget and results of online preliminary contests, each team will go to 1 or 2 contests + ICPC EC Final.
- We usually depart on Friday and return on Monday by plane.

# Contest plan

- Nov 16 - 17
  - 1 team goes to **Shenyang** for ICPC Asia Regional Contest
  - Maybe 1 team goes to Beijing for CCPC Final
- Nov 23 - 24
  - 3 teams go to **Shanghai** for ICPC Asia Regional Contest
- Nov 30 - Dec 1
  - 2 teams go to ICPC Asia **Hong Kong** Regional Contest
- Dec 14 - 15
  - Qualified teams go to **Xi'an** for ICPC EC Final

# ICPC online preliminary contests

- Shenyang: Sep 14; Shanghai: Sep 15. 12:00 – 17:00.
- Everyone is welcomed; top 9 students are encouraged.

# Selection contest

- Planned on **September 12 (Thu), 7:00 pm - 9:30 pm.**
  - Individual contest.
  - Rank by ICPC rules.
  - Printed/written notes are allowed.
  - Contest environment: PC<sup>2</sup>, <https://pc2.ecs.csus.edu/>
- Contestants ranking **top 9** will be qualified to represent HKU to participate in the contests.
- Practice contest: September 10 (Tue), 7:00 pm – 8:00 pm.
  - Not mandatory.

Practice and training

# Being self-motivated

- You join us because you are interested in programming contests, and you can leave at any point if you are not any more.
- We assist you to participate in the contests, including organizing online training sessions.
- However, we do not and cannot force you to do anything. You have to be **self-motivated**.

# Your benefits

- Opportunity to participate in ACM-ICPC and CCPC.
- The department appoints a TA (Bintao SUN) for you to consult, regarding the competition and solving programming problems.
- Good prize helps in finding a good job in industry.
- Meet new friends that have the same interest with you.



# Practice

- Online judge
- Codeforces: <https://codeforces.com/>
- Peking University Online Judge (many classic problems, some are a little bit outdated): <http://poj.org/>
- ACM-ICPC Live Archive (past contest problems): <https://icpcarchive.ecs.baylor.edu/>
- Etc.
- Virtual judge (DIY contests): <https://vjudge.net/>

# Training plan – this semester

- We organize practice contests regularly, aiming for this year's regional contests.
- Team contest. Follow ICPC rules.
- Problems selected from past regional contests.
- If you are new, you can find some resources in <http://i.cs.hku.hk/~provinci/training.html>

# Training plan – next semester

- Mainly for beginners.
- We collect some online problems to talk about algorithmic topics.

# Consultation

- Send me an email if you have any questions about the contest, both technical and non-technical.
- Bintaο SUN, [btsun@cs.hku.hk](mailto:btsun@cs.hku.hk)
- Appoint with me if you need face to face consultation.
- Our website: <https://i.cs.hku.hk/~provinci/>

Any questions?