

ACM ICPC Boot Camp

Ankesh Gupta

IIT Delhi

Credits: Adapted from IIT Bombay Slides

January 17, 2019

What is ACM ICPC?

- ICPC stands for Inter-Collegiate Programming Contest, conducted by ACM
- It is the most prestigious college level programming contest
- 3 stages - Online, Onsite(regionals), World Finals
- Online round has 3500 teams participating
- India conducts 4-5 regionals(this year 4)

	Amritapuri	NITTTR Kolkata + UIET Kanpur	IIT, Kharagpur	IIITM, Gwalior + IIIT Pune
Registration Link	Click here	Click here	Click here	Click here
Online Contest Date	23rd October 2018, 19:30 IST to 22:30 IST			
Online Contest Link	CodeChef			
Last Date for Registration of Online Contest	18th October 2018	13th October 2018	14th October 2018	16th October 2018
Onsite Contest Date	28th - 29th December 2018	21st - 22nd December 2018	15th - 16th December 2018	17th - 18th December 2018
Onsite Round Platform	To Be Announced			
Onsite Slots (Tentatively)	225	160	85	125
World Finals Slots (Last year)	NA	NA	NA	NA

How to Participate?

- Teams of 3.
- Each team should have a coach, who must be a professor.
- The coach registers the team for the online round.
- Typically 2-3 teams qualify from the institute per regional after online round.
- Preference given to maximise number of distinct colleges at a regional.

A Typical Contest Problem

- Problem
- Input and Output format
- Number of instances
- Variable bounds
- Constraints
 - Time limit
 - Space limit

Input/Output Format

- Input is from STDIN and output is to be made to STDOUT
- For C++: scanf and printf are in general faster than cin and cout, so it is recommended that you use these for i/o handling.
- General Tip: Time for i/o depends on the number of system calls made, so for instance if you have to take as input 'n' numbers, separated by space, it is useful to take the input as a single line and then parse it to get 'n' numbers rather than making 'n' different system calls.
- Make sure that you follow the output format and avoid trivial wrong submissions only because of formatting the output incorrectly.

- O-notation :
 $f(n) = O(g(n))$ if $f(n)$ is asymptotically less than some multiple of $g(n)$.
- Generally helpful to estimate complexity of algorithms.
- For example,
$$f(n) = 2n^2 + 4n^3 = O(n^3)$$

Variable Bounds

- A typical CPU is assumed to process 10^8 instructions in a second.
- This assumption will help us deciding the targetted order of the algorithm while inspecting the input variable constraints.
- A brief idea is as follows: Divide 10^8 by the number of test cases and then use the following table to predict order.

Variable Bound * No. Test Cases	Predicted Order
10^8	$O(n)$
10^5	$O(n \log n)$
10^4	$O(n^2)$
10^{16}	$O(\sqrt{n}), O(n \log n)$

Data Ranges

- It is important that you understand which data type to use.
- For this it is necessary to have a feel of the ranges of various data types.
- This avoids loss of time during the competition in figuring out the appropriate data type say long int or just int or long long int.
- The foll. table gives a brief idea about some data types. You can google for other data types.

Data Type	Range
short	32,768 to 32,767
int	2,147,483,648 to 2,147,483,647
long long	9,223,372,036,854,775,808 - 9,223,372,036,854,775,807

Types of errors

- Compilation Error
- Time Limit Exceeded
- Memory Limit Exceeded
- Wrong Answer
- You should be familiar with these errors before attempting a contest
- Appropriate penalties are put for every erroneous solution

Programming Style

- C, C++, Java is allowed in all contests. So preferably practice in only these
- Using templates can ease your job during a contest
- Be aware! Corner cases can get tricky!
- Be careful about the output format; take care in removing stray outputs.

- Using STL classes/functions can speed up your work.
- They may be more efficient than your own implementation.
- Some STL classes that help,
vector, list, queue, set, map, deque, priority queue
- STL functions that help,
sort, next permutation, etc...

- Algorithms library is a very important library with many predefined functions which can be used directly and inturn save time.
- Some important functions are:
 - Merge
 - Sort
 - Min/Max
 - Lexographical Compare
 - Make Binary Heap
 - Modifying/Nonmodifying Sequence Operations

Where and How to Practice?

- Codeforces div3, div2, educational rounds, div1
- Codechef cookoff (monthly contest) and long contests should be tried
- SPOJ solutions are not easy to find but lots of discussion available online for same
- Hackerearth, Hackerrank - simple platform with a good interface and problems escalating from simple to difficult

Success is team work

- Knowing who does what best can help in planning
- Try forming a team with people having different strengths
- Cross check your teammates solution for higher accuracy before submitting
- Ensure that all three are not stuck on the same problem. Start attacking different problems
- Stay cool! Dont panic! :)

Thank You!