ACM ICPC & OTHER HINTS TIPS TRICKS

Oleg Šelajev
@sheljev
AGENDA

> QUICK RECAP
> ACQUIRE THE TASTE
> ACTUAL ADVICE
> SAMPLE ROUND
COMPETITIONS

> PROBLEM SOLVING
> NO SOFTWARE ENGINEERING
> TEAM WORK (USUALLY)
ACM ICPC

> ANNUAL, STAGED

> STUDENTS, AGE CAP

> 5H, 1C, 3P, ~12T

> AT MOST 5X OR 2 FINALS
IEEEEXTREME

> ANNUAL

> IEEE MEMBERS? :)

> 24H, 3C, 3P, ~24T
TOPCODER

> ALL THE TIME
> RATING SYSTEM
> 1.5H, 1C, 1P, 3T
> CHALLENGE PHASE
This is problem solving:

I don't like the way we are doing it.

What way would you like to do it?

We could try this other way—how about that?

We could try this way—how about that?

We could forget the whole thing—how about that?
TASTE THE BLOOD

> http://goo.gl/qjzJUE

> let’s form teams!

> 15 minutes
Engineering Flowchart

DOES IT MOVE?

No

Should it?

No

No Problem

Yes

WD-40

Yes

No Problem

Yes

Should it?

No

No Problem

Yes

No

No

No Problem
PREPARATION ROUND

> CAN YOU ADD NUMBERS?

> CAN YOU ADD MORE NUMBERS?
PREPARATION ROUND

> TIME LIMITS
> API SUBTLETIES
> MEMORY LIMITS
> EXCEPTION REPORTING
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>00:21</td>
<td>+</td>
<td>+1</td>
<td>01:31</td>
<td>01:14</td>
<td>01:36</td>
<td>00:23</td>
<td>00:32</td>
<td>+</td>
<td>01:21</td>
<td>+</td>
</tr>
<tr>
<td>-1</td>
<td>00:16</td>
<td>-1</td>
<td>+</td>
<td>00:58</td>
<td>+</td>
<td>01:36</td>
<td>00:07</td>
<td>00:22</td>
<td>+</td>
<td>00:45</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:19</td>
<td>+</td>
<td>+2</td>
<td>01:14</td>
<td>01:14</td>
<td>+</td>
<td>00:23</td>
<td>00:32</td>
<td>-1</td>
<td>00:48</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:27</td>
<td>+</td>
<td>+3</td>
<td>01:45</td>
<td>+</td>
<td>+</td>
<td>00:13</td>
<td>00:09</td>
<td>+</td>
<td>01:10</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:11</td>
<td>+</td>
<td>+4</td>
<td>01:10</td>
<td>01:25</td>
<td>+</td>
<td>00:16</td>
<td>00:23</td>
<td>+</td>
<td>01:39</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:48</td>
<td>+</td>
<td>+1</td>
<td>01:14</td>
<td>01:23</td>
<td>+</td>
<td>00:40</td>
<td>+</td>
<td>+2</td>
<td>01:43</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:36</td>
<td>-2</td>
<td>+</td>
<td>00:59</td>
<td>01:36</td>
<td>00:23</td>
<td>00:25</td>
<td>00:30</td>
<td>+</td>
<td>01:11</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:04</td>
<td>-2</td>
<td>+</td>
<td>00:19</td>
<td>00:25</td>
<td>00:17</td>
<td>00:07</td>
<td>00:29</td>
<td>+</td>
<td>01:15</td>
<td>+</td>
</tr>
<tr>
<td>+</td>
<td>00:25</td>
<td>+</td>
<td>+1</td>
<td>01:46</td>
<td>+</td>
<td>00:14</td>
<td>00:07</td>
<td>00:58</td>
<td>+</td>
<td>+1</td>
<td>01:32</td>
</tr>
<tr>
<td>+</td>
<td>00:15</td>
<td>+</td>
<td>+1</td>
<td>01:16</td>
<td>01:16</td>
<td>00:22</td>
<td>00:49</td>
<td>+</td>
<td>+1</td>
<td>01:32</td>
<td>+</td>
</tr>
</tbody>
</table>
RESPONSIBILITIES

- SOLVING ON PAPER
- PAIR PROGRAMMING
- MAKING COFFEE
- ASKING DUMB QUESTIONS
- PREPARING TEST CASES
Compilation error

Parse error. Internet Explorer has a non-standard interpretation of trailing commas. Arrays will have the wrong length and objects will not parse at all.

In /Users/erwan/Devel/Play20/samples/jscompilation/app/assets/javascripts/test.js at line 9.

5 function tata(toto, second) {
6     alert(sum(first, second));
7 }
8
9 tata([2,3,], 4);
### Searching

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Data Structure</th>
<th>Time Complexity</th>
<th>Space Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depth First Search (DFS)</td>
<td>Graph of (</td>
<td>V</td>
<td>) vertices and (</td>
</tr>
<tr>
<td>Breadth First Search (BFS)</td>
<td>Graph of (</td>
<td>V</td>
<td>) vertices and (</td>
</tr>
<tr>
<td>Binary search</td>
<td>Sorted array of (n) elements</td>
<td>(O(\log(n)))</td>
<td>(O(\log(n)))</td>
</tr>
<tr>
<td>Linear (Brute Force)</td>
<td>Array</td>
<td>(O(n))</td>
<td>(O(n))</td>
</tr>
<tr>
<td>Shortest path by Dijkstra, using a Min-heap as priority queue</td>
<td>Graph with (</td>
<td>V</td>
<td>) vertices and (</td>
</tr>
<tr>
<td>Shortest path by Dijkstra, using an unsorted array as priority queue</td>
<td>Graph with (</td>
<td>V</td>
<td>) vertices and (</td>
</tr>
<tr>
<td>Shortest path by Bellman-Ford</td>
<td>Graph with (</td>
<td>V</td>
<td>) vertices and (</td>
</tr>
</tbody>
</table>
TRAINING

> TRAINING IS HARD
> TRAINING IS TIME CONSUMING
> PEOPLE ARE LAZY
> DON'T BE LAZY!
> PROFIT!
> http://codeforces.com/gym/100503
http://codeforces.com/gym/100503

REGISTER

LOGIN

SOLVE
DON'T WORRY SIR,

I'M FROM THE INTERNET.