

# Contestant Handbook

## Conduct of the Contest

The duration of the contest round is **five hours**. During the round each team is provided with one personal computer and a set of problems. At least eight problems and at most fourteen problems will be proposed.

During the Contest, contestants solve the proposed problems. A solution to a problem is a program written in one of the supported programming languages. Contestants may solve different problems using different supported programming languages.

Contestants may bring and use unannotated natural language dictionaries (except electronic ones), blank sheets of paper and instruments for writing **only**. Contestants **may not** bring or use any books (except dictionaries), reference manuals, electronic dictionaries, program listings, any machine-readable information (software or data on any kind of storage), computing devices (handhelds, portable PCs, notebooks, calculators, smart watches), mobile phones, or any other communication devices.

During the round, contestants may communicate to members of their team, members of the Executive Committee of the Jury, the Technical Committee, and the Support Staff **only**.

Before the beginning of the Contest, all computers will be turned on. During the Contest each team will be provided with an envelope containing problem statements (3 copies), limits and evaluation environment, and the problem statements errata and clarifications. The envelope is placed under the keyboard.

Contestants **may not** touch the computer or problem statements before the beginning of the Contest. The Contest will begin after the notification “*THE CONTEST IS STARTED*”.

Contestants may use network printer during the Contest. Support Staff delivers printouts to the teams.

In case of any issues with the computers, printers, or anything else contestants should ask the Support Staff for help.

## Run evaluation

*Run* is a solution to a problem submitted for judging. The size of the source file with the run may not exceed 256KiB.

Immediately after submission of any run, the team may continue working on other problems.

Contest software evaluates each run and marks it as *accepted* or *rejected*.

The run is evaluated by executing it on a secret set of tests, common for all contestants. A run is accepted only if it gives correct answers to all tests.

The *memory limit* is the maximum amount of memory that a run may utilize on each test. The *time limit* is the maximum execution time per test. The time and memory limits for problems are provided with the problem statements. The run is not accepted if the program exceeds these limits.

As soon as the run is evaluated, contest software displays evaluation results. The team is informed whether the run is accepted or not. If the run is rejected, the error type and the test number (when applicable) are indicated.

All test cases are numbered starting from one. The first test cases in the test set are the same as the sample tests from the problem statement. The following tests are ordered with the idea to make easier test cases come before harder ones, although there are no guarantees in that respect.

The possible outcomes are listed in the following table.

Outcome	Test Number	Comment	Possible Reasons
Compilation error	No	Executable file was not created after compilation.	<ul style="list-style-type: none"><li>• Syntax error in the program;</li><li>• wrong file extension or language specified.</li></ul>
Security violation	Yes	The program tried to violate the Contest rules.	<ul style="list-style-type: none"><li>• Error in the program;</li><li>• purposeful rules violation (the violating team is disqualified in this case).</li></ul>
Runtime error	Yes	The program terminates with non-zero exit code or throws an uncaught OS exception.	<ul style="list-style-type: none"><li>• Runtime error;</li><li>• uncaught exception;</li><li>• missing <code>'return 0'</code> statement in C++ main function;</li><li>• <code>'return (non-zero)'</code> statement in C++ main function;</li><li>• <code>'System.exit(non-zero)'</code> in Java.</li></ul>
Time limit exceeded	Yes	The program exceeds the time limit.	<ul style="list-style-type: none"><li>• Inefficient solution;</li><li>• error in the program.</li></ul>
Memory limit exceeded	Yes	The program exceeds the memory limit.	<ul style="list-style-type: none"><li>• Inefficient solution;</li><li>• error in the program.</li></ul>
Idleness limit exceeded	Yes	The program does not consume processor time for a long period.	<ul style="list-style-type: none"><li>• Input from console;</li><li>• not flushing output in interactive problem, waiting for input while output is still buffered;</li><li>• error in protocol in interactive problem, mistakenly waiting for input;</li><li>• error in the program.</li></ul>
Presentation error	Yes	The checker cannot check output because it does not match the format specified in the problem statement.	<ul style="list-style-type: none"><li>• Output format is not correct;</li><li>• no output file;</li><li>• wrong output file name.</li></ul>
Wrong answer	Yes	The answer is not correct.	The algorithm is not correct.
Accepted	No	Run is accepted.	Program is correct.

The possible outcomes in the table are listed in their order of priority. For example, if runtime error has occurred, then output is not checked.

Evaluation process may be stopped several minutes before the end of the Contest. All runs submitted after this moment will be evaluated just after the end of the Contest.

Runs are evaluated on *AMD Phenom II 3.20GHz* computers under *Windows 7, Service Pack 1*.

Runs are not allowed to:

- access the network;
- perform any I/O except for opening, closing, reading, and writing files and standard streams that are explicitly specified in the problem statement;
- execute other programs and create new processes;
- work with subdirectories;
- create or manipulate any GUI resources (windows, dialog boxes, etc.);
- work with external devices (sound, printer, etc.);
- attack system security;
- do anything else that can stir the evaluation process and the Contest.

## Programming languages

A solution to a problem is a program written in one of the following programming languages:

Language	Compilation command
Java 8 Update 102	<code>javac &lt;source file&gt;</code>
Visual C++ 2013	<code>cl /O2 /TP /EHs &lt;source file&gt;</code>
GNU C++ (MinGW) 6.2	<code>g++ -O2 -Wl,--stack=67108864 -x c++ -std=c++14 &lt;source file&gt;</code>
Python 3.5.2	Not compiled

## Computer Configuration

Each team is supplied with one computer with installed operating system Windows 7, connected to the local network. Your Windows login name is “**neerc**”, and your password is “**contest**”.

The folder `c:\soft` contains the following software:

Software	Folder
Visual Studio Express 2013	<code>vs-2013</code>
MinGW (GNU C++ 6.2.0.1)	<code>mingw</code>
Java 8 SDK Update 102	<code>java</code>
IntelliJ IDEA CE 2016.2	<code>idea</code>
Eclipse 4.6 (JDT + CDT)	<code>eclipse</code>
Python 3.5.2	<code>python</code>
PyCharm CE 2016.2	<code>pycharm</code>
WingIDE 101 5.1.12	<code>wingide</code>
Code::Blocks 16.01	<code>codeblocks</code>
Far Manager 3.0	<code>far</code>
gViM 7.4	<code>vim</code>

Paths to compilers are included in the environment variable `PATH`.

You will be provided with electronic version of the problem statements and sample input and output files in `c:\work` folder.

You **may not** store your solutions and other files outside the `c:\work` folder.

## Clarification Requests

A contestant may submit a claim of ambiguity or error in a problem statement by submitting a clarification request. Clarification requests are accepted only in English.

If the Jury agrees that an ambiguity or error exists, a clarification will be issued to all contestants. The Jury encourages contestants to use the sample input and output for resolving apparent ambiguities.

## Scoring of a Contest

Teams are ranked according to the most problems solved. Teams solving the same number of problems are ranked by the least total time and, if needed, by the earliest submit time of the last accepted run.

The *total time* is the sum of the penalty time for each problem solved. The *penalty time* 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 **successfully compiled, but rejected** run for that problem before the first accepted run. There is no penalty time for a problem that is not solved.

## Practice Session

During the practice session teams become familiar with the contest environment and the contest software solving sample problems. During the practice session teams **may not** store any source code anywhere

except working directory and **may not** attach any devices to the computer or alter its hardware configuration. The results of the practice session are not taken into consideration when determining the Contest standings. However, the Executive Committee of the Jury may disqualify any team violating the contest rules.

## PCMS Web Client User Guide

To start a *PCMS Web Client*, double click on *PCMS Client* icon. A browser opens the *Information* page.

The *Information* page contains contest information and messages sent by the Jury. During the Contest you will receive messages from the Jury with the results of your runs and clarifications. Incoming messages will be listed on this page. You can always read all your previous messages in this list.

The *Monitor* page contains the standings table. Teams are displayed as the rows of the table, ordered by team's rank. Problems correspond to the table columns. The intersection of team and problem contains information about team's result for that problem. Possible values are:

- “.” — team has no runs for this problem;
- “+” — team has solved this problem, the first run was successful;
- “+k” — team has solved this problem after  $k$  unsuccessful runs;
- “-k” — team has  $k$  unsuccessful runs for this problem.

The time under this value is the time of the first accepted run on this problem, measured in minutes.

To submit a program for evaluation, you should select the *Submit* page. Choose the problem you have solved from the *Problem* combo box and the language of your solution from the *Language* combo box. Press the *Choose File* button and choose the file that contains your solution, then press the *Submit solution* button. You will see a page, confirming that your solution was successfully sent to the server for evaluation. Your solution will appear in the *Runs* page. The evaluation result will appear on this page when the solution is evaluated. You may solve other problems while waiting for the result.

To submit a clarification request, you should select the *Questions* page. Choose the problem you have the clarification request for from the *Problem* combo box and type your clarification request into the *Question* field, then press the *Ask a question* button. The list of already asked questions and corresponding answers is displayed on the same page.

## Java Tips and Tricks

Before starting Eclipse copy workspace from `c:\eclipse\workspace` to your working directory. In the other case you will not be able to use external help files.

Your solutions will be executed using the command `“java -Xmx<memory limit> -Xss64m <class file>”`.

The stack size of all threads created by your program is set to 64MiB. In Java 8 the correct stack size is set even for the main thread of the program.

The first class defined in your solution must be declared `public` and must contain the method `main`. Otherwise, you will receive *Compilation Error* outcome.

`Scanner` class is slow. You can use `BufferedReader` and `StreamTokenizer` classes instead.

Before using `Scanner`, `PrintWriter` and other classes that read or write floating-point numbers include the following line in your code: `“Locale.setDefault(Locale.US);”`.