Languages (1)

Java (myjava, myjavac)

```
java -Xrs -Xss8m -Xmx1272864k
javac -encoding UTF-8 -d . "$@"
```

C-executable

```
gcc -x c -Wall -O2 -static -pipe -o "$DEST" "$@" -lm
```

C++

```
g++ -x c++ -Wall -O2 -static -pipe -o "$DEST" "$@"
```

Languages (2)

Python

pypy3

C# ***(Company teams only)***

mono

kotlin

kotlinc -d . **(compile)**

kotlin -Dfile.encoding=UTF-8 -J-XX:+UseSerialGC

-J-Xss${MEMSTACK}k -J-Xms${MEMLIMITJAVA}k

-J-Xmx${MEMLIMITJAVA}k '$MAINCLASS' "\$@" **(run)**

Miscellaneous

- Read input from STDIN
- Write output to STDOUT

- Java main method is detected automatically

- Only use your own team computer

- Questions about the problem set: **Send a clarification request**
- All other questions: **ask CHipCie for help via Discord #questions!**

Score

- Team with most correct submissions within the *least* time wins!
- Correct submission will yield points equal to minutes passed.
- Every incorrect submission yields 20 minutes penalty when submitting the correct solution to a problem.

Interactive problems (1)

- This year, the contest includes interactive problems.
- For each of these problems, the jury provides a program to test your solution locally.
- You can find this tool in the samples.zip provided by the organisation, next to the samples of the other problems.
- Each of these is called `testing_tool.py`, and starts with an explanation on how to use it.

Interactive problems (2)

- Note that these tools only test the basic interaction of your program. If your program works locally, it may still fail when submitting it in DOMjudge.
- These tools need **Python 3** to run them. Please make sure you can use the testing tool for problem X during the test session.
- Note that your submission should flush its standard output after each write, or else you may get a TIME_LIMIT_EXCEEDED.
- We guarantee that each problem is solvable in all of C++, Java, and Python, but *note that slower languages may require more efficient code*.