



Regional Programming Environment

Last Updated: September 1, 2026

This page describes the current plans for the ICPC Greater New York Regional Programming Environment which will be available to each team. Please note that ***these plans are subject to change***.

Each team at the Greater New York Regional contest will be allowed to use one laptop, shared between team members. Each team is required to bring their own laptop that is capable of successfully running the Greater New York Regional contest image (which is essentially the image that will be used at the NAC and World Finals in 2027). You **must provide the manufacturer and model number and web link** of the laptop you plan to use at the contest to the [Greater New York System's Team](#) at least two weeks prior to the contest date or we may not be able to provide technical support for you on contest day.

Hardware

- **Laptop**

- The Greater New York Regional contest image supports most modern Intel x64 (or compatible) based laptops. It is the team's responsibility to ensure that the contest image works on the laptop they intend to use for the contest. We will be using WiFi at the on-site contest, so be sure you learn how to configure the WiFi on the contest image.
- While we **do not** recommend it, Apple MacBook Pro's using the M-series silicon may be used as a host for an x64 based virtual machine using [UTM](#) (or equivalent). The contest image does run successfully on a virtual machine on a MacBook Pro, however, the performance is somewhat degraded since the x64 hardware is being emulated on the M-series processors. Note that the MacBook Pro using the M1 series Apple chip may be too slow to use for the contest. <https://image.icpc.global/gny2026pr/doc/MacBookProImageSetup.pdf> provides detailed instructions on how to set up and run the contest image on a MacBook Pro M-series laptop. While we will try to help you get the image working on your MacBook Pro, we are unable to guarantee that we can make it work. **It is extremely unlikely that we will be able to offer any assistance at all on contest day for teams using a MacBook Pro.** It is therefore prudent to fully test your MacBook Pro system prior to the contest and to try **BOTH** of the setup options described in the document linked above (with and without a thumb drive). We **strongly** recommend that you find an Intel x64 based PC instead of using a MacBook Pro and fully test that prior to arrival on contest day.
- You will **NOT** be permitted to use VirtualBox, or any other virtual environment on Intel x64 (or compatible) based laptops at the contest. You will be supplied with a bootable thumb drive containing the contest image, and you must boot of that if using an Intel x64 (or compatible) based laptop.

- **External screen:**

- You are not allowed to use an external monitor at the contest site. Only the built-in monitor on the laptop is allowed.

- **External keyboard:**
 - You are permitted to bring your own *wired* external keyboard and use it on contest day. **At least two weeks prior to the contest date, you must supply us with the manufacturer, model number and manufacturer’s web page link for the keyboard you intend to use so that we can approve it.** You can send the required information to us using [this link](#). We may not allow certain keyboards at the discretion of the Regional Contest Director and the system’s team. For example, so-called “smart” or “programmable” keyboard will *not* be permitted. Wireless keyboards will **not** be permitted. **If you do not get pre-approval for an external keyboard, you will not be allowed to use it during the contest.**
- **External Mouse:**
 - You are permitted to bring your own external mouse and use it on contest day. **At least two weeks prior to the contest date, you must supply us with the manufacturer, model number and manufacturer’s web page link for the mouse you intend to use so that we can approve it.** You can send the required information to us using [this link](#). We may not allow a specific mouse at the discretion of the Regional Contest Director and the system’s team. For example, a so-called “smart” or “programmable” mouse will *not* be permitted. Wireless mice will **not** be permitted. **If you do not get pre-approval for an external mouse, you will not be allowed to use it during the contest.**

Software

The software configuration for the Greater New York regional contest will consist of the following:

- **OS:**
 - Ubuntu 26.04 LTS Linux (64-bit) with Linux kernel 7.0.0hwe. A list of packages included in the OS image can be found [here](#).
- **Desktop:**
 - XFCE4
- **Editors**
 - vi/vim
 - gvim
 - emacs
 - gedit
 - geany
 - kate
- **Languages:**
 - **Java** (Openjdk version 25.0.3)
 - compiler flags: `-encoding UTF-8 -sourcepath . -d . {files}`
 - runtime flags: `-Dfile.encoding=UTF-8 -XX:+UseSerialGC -Xss64m -Xms{memlim}m -Xmx{memlim}m`
 - **C** (gcc version 15.2.0)
 - compiler flags: `-x c -g -O2 -std=gnu23 -static {files} -lm`

- **C++** (g++ version 15.2.0)
 - compiler flags: `-x c++ -g -O2 -std=gnu++23 -static {files}`
- **Python 3** (PyPy3 version [7.3.23](#) providing Python 3.11.15) A list of the installed Python modules is available [here](#).
- **Kotlin** (Version 2.4.10-release-377)
 - compiler flags: `-d . {files}`
 - runtime flags: `-Dfile.encoding=UTF-8 -J-XX:+UseSerialGC -J-Xss64m -J-Xms{memlim}m -J-Xmx{memlim}m`
- **IDEs:**
 - **Eclipse** (version 4.40.0 (2026-06)), configured with:
 - Java Development Tooling (JDT) version 3.20.600.v20260601-0713 using Java as listed above.
 - C++ Development Tooling (CDT) version 12.5.0 202606022100 using C++ as listed above.
 - PyDev Python Development Tooling version 13.1.0.202509210817 using Python3 as listed above.
 - **IntelliJ IDEA** (Community Edition, version 2026.2.1), configured with:
 - Java as listed above
 - Kotlin as listed above
 - **CLion** (version 2026.2.1), configured with:
 - C/C++ as listed above
 - **Pycharm** (Community Edition version 2026.2.1), configured with:
 - Python 3 as listed above
 - **Code::Blocks** (version 25.03), configured with:
 - C/C++ as listed above
 - **Visual Studio Code**(version 1.132.0), configured with:
 - C/C++ as listed above
 - NOTE: The Judges will compile and execute C/C++ programs using C/C++ as listed under **Languages**, above, NOT using Microsoft C/C++.

Compilation of Submissions

During the contest, teams will submit proposed solutions to the contest problems to the Judges using the [PC²](#) contest control system (CCS). Check the [PC² Team Guide](#) for further details. Source files submitted to the Judges will be compiled using the language systems and compiler flags described above.

The “**{files}**” notations in the descriptions above represents the list of source files from the submission which will actually be compiled. Files with the following suffixes (and only files with these suffixes) will be submitted to the compiler:

- For C submissions: files ending with `.c`
- For C++ submissions: files ending with `.cc`, `.cpp`, `.cxx`, or `.c++`
- For Java submissions: files ending with `.java`
- For Python submissions: files ending with `.py`
- For Kotlin submissions: files ending with `.kt`

Execution of Submissions

For each language, if the above compilation step is successful then the submission will be executed as follows:

- For C/C++: the executable file generated by the compiler will be executed to generate the output of the submission.
- For Python 3: the main source file will be executed by the PyPy3 Python3 interpreter to generate the output of the submission.
- For Java: the compiled main class will be executed using the **java** command with the runtime flags listed above.
- For Kotlin: the compiled main class will be executed using the **kotlin** command with the runtime flags listed above.

Compilation and execution as described above will take place in a “sandbox” on a dedicated judging machine. The judging machines (likely *Amazon Web Services EC2 m5.xlarge* instances) will be different from your laptop, so timing results may vary from yours. In addition, we **strongly** discourage the use of any `#pragma`’s in your C/C++ code due to the potential hardware differences. If you choose to go against our recommendation, and decide to use `#pragma`’s in your code, you do so at your own risk. The sandbox will allocate memory according to the memory limit specified in the problem statement; the entire program, including its runtime environment, must execute within this memory limit. For interpreted languages (Java, Python, and Kotlin) the runtime environment includes the interpreter (that is, the JVM for Java/Kotlin and the Python interpreter for Python).

The sandbox memory allocation size will be the same for all languages.

Reference Materials

The following packages will be available on Greater New York contest image:

- JDK JavaDocs
- C++ STL docs
- Kotlin docs
- Python docs
- PC² Team Guide (as linked above)
- Additional Information:
 - [Judging Notes](#)
 - [Technical Notes](#)