



Introduction

The PC² Contest Control System (CCS) provides multiple ways for teams to participate in a programming contest. The purpose of this Team Guide is to explain how each of these facilities works.

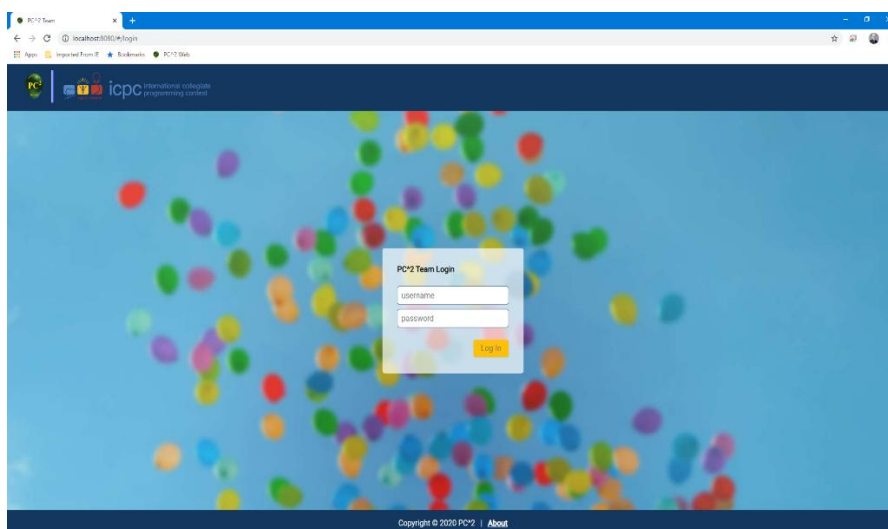
Most teams prefer the PC² Web Team Interface (WTI), which provides a browser-based facility – meaning that once the Contest Administrator has set up the PC² system properly, all a team needs to participate in the contest is a web browser. The WTI supports a complete set of contest operations for teams, including the ability to send submissions (problem solutions) and clarification requests to the judges, the ability to examine the current contest scoreboard, and the ability to see the current “contest clock”. The WTI is described in the following section.

Some teams prefer the alternative of a *command-line* interface. PC² supports this via a command-line tool called **pc2submit**, which is described in a subsequent section.¹

The PC² Web Team Interface (WTI)

Logging into PC²

To connect to PC² using a browser, enter the URL of the PC² Web Team Interface into the address bar of your browser. (Note that your Contest Director may have arranged that there are links for PC² on the browser shortcut bar, the desktop, or on a desktop menu; if not, the correct URL can be obtained from the Contest Director). When connected to the PC² WTI, the following *Login* screen will appear:



To login to PC²:

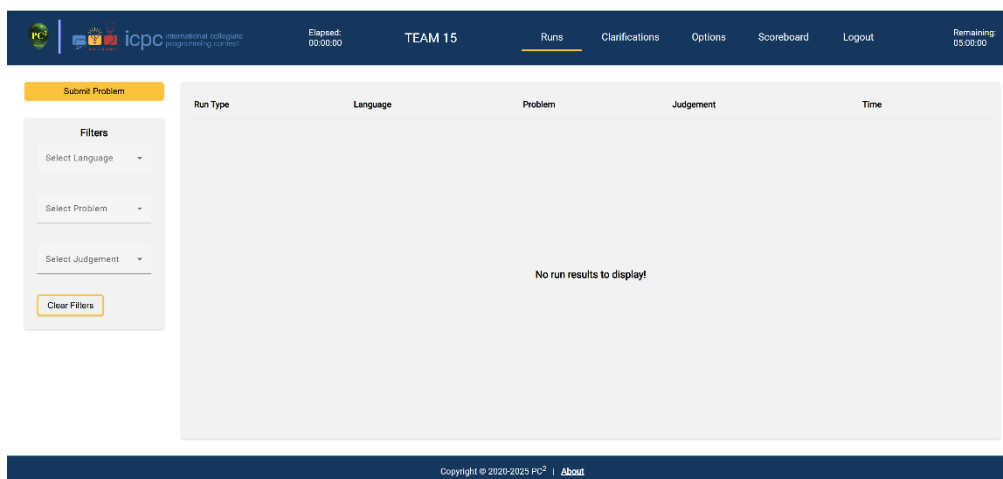
- Click on the **Username** box and enter your assigned team ID. Your team ID will be of the form teamXX, where XX is your assigned team number; for example, “team2” or “team15”.
- Click on the **Password** box and enter your assigned password.

¹ There also exists an older team interface option referred to as the PC² “Team Thick Client”. This option is described in a separate document which can be found by downloading the PC² system from <https://pc2ccs.github.io>. However, the Team Thick Client requires installing a complete PC² system on each team machine and is not described in this Team Guide.



- Click on the **Login** button.

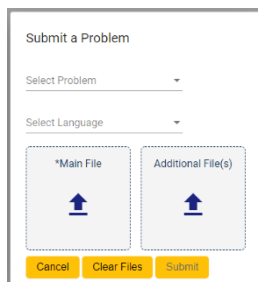
Successfully logging in will display a page like the following (called the **Runs** page):



Submitting a Problem for Judging

To submit a problem to be judged:

- Click on the **Runs** tab at the top of the page (if not already on the **Runs** page).
- Click on the **Submit Problem** button in the upper left hand corner of the **Runs** page. A pop-up with the title “Submit a Problem” will display, as shown here:



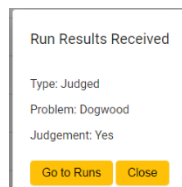
- Select the problem for which you are submitting a solution from the drop down list titled **Select Problem**.
- Select the language that your solution uses from the drop down list titled **Select Language**.
- Select the main file that contains your solution. To do this, click on the  symbol in the box titled **Main File**. This will display a file dialog which will allow you to locate and select your main file. Select only source code files for submission (do not submit data files or executable files).
- If additional source code files are needed for your solution, you can click on the  in the box titled **Additional File(s)**. This will display another file dialog which will allow you to locate and select your additional file(s).
- To remove the currently-selected files from the **Main File** and **Additional File(s)** list, click the **Clear Files** button.



- Once you have made the desired selections, click the **Submit** button to submit your solution to be judged. Note: your submission must include at least a selected problem, a selected language, and a selected main file. The **Submit** button will be disabled until these items have all been selected.
- If you decide that you do not want to submit your solution at this time, click the **Cancel** button to be returned to the Runs page.

Viewing Submission Results

When a submission has been judged, the team will receive a “pop-up notification”² like the following:



To view the status of *all* submissions for your team, click on the **Runs** tab at the top of the page. This will display the **Runs** page, which will look like the following:

The screenshot shows the "Runs" page of the PC² contest control system. At the top, there is a navigation bar with the PC² logo, "icpc international college programming contest", "Elapsed: 00:18:55", "TEAM 15", and tabs for "Runs", "Clarifications", "Options", "Scoreboard", and "Logout". A "Remaining: 04:41:05" timer is also present. On the left, there is a "Submit Problem" button and a "Filters" section with dropdown menus for "Select Language", "Select Problem", and "Select Judgement", along with a "Clear Filters" button. The main area contains a table of submissions.

Run ID	Language	Problem	Judgement	Time
1-5	Python	Sumit	No - Wrong Answer	14
1-4	GNU C++	Sumit	No - Wrong Answer	10
1-3	Java	Hello	Yes	5
1-2	GNU C++	Sumit	No - Time Limit Exceeded	2
1-1	Java	Hello	No - Run-time Error	1

At the bottom of the page, there is a footer: "Copyright © 2020-2025 PC² | About".

- The **Runs** page contains a table that displays all submissions. Submission information includes the **Run ID** (comprised of the *Site* and the *Run Number*, separated by a dash), the **Language** the submission was written in, the **Problem** for which the solution was submitted, the **Judgement** received for the submission, and the **Time** the run was submitted (in minutes, relative to the start of the contest).
 - If the judgement is *Preliminary* (see below) it will be labeled as such in the table.
 - If no judgement has yet been determined, the **Judgement** field will be empty.
- There are three **filters** that can be used to filter (limit) the submissions being displayed in the **Runs** table: *Language*, *Problem*, and *Judgement*.
 - If you want to filter runs so that *only runs submitted for a single language* are shown, click on the **Select Language** drop down list and select the language you want to filter by.

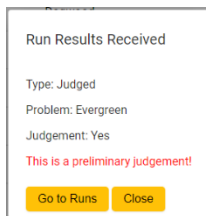
² Pop-ups like this can be disabled if desired; see **Options** below.



- If you want to filter runs so that *only runs submitted for a single problem* are shown, click on the **Select Problem** drop down list and select the problem you want to filter by.
 - If you want to filter runs so that *only runs which received a specific judgement* are shown, click on the **Select Judgement** drop down list and select the judgement you want to filter by.
 - Filters can be used in combination. To clear all filter selections, click the **Clear Filters** button.
- Preliminary Judgements:

The Contest Director can choose to configure PC² to judge problems either automatically (“by computer”, a process called *AutoJudging*), or manually (by human judges), or *both*.

If a problem is being judged both using AutoJudging *and* manually, the system will return the AutoJudging result to the team as soon as it is finished. This judgement is termed a *Preliminary Judgement* and will be so-labeled in the **Runs** table. The Run Result pop-up notification for Preliminary Judgements (if not disabled) will display an appropriate “Preliminary Judgement” notice in red font, as shown below:



When the judges manually check the submission result and choose a final judgement, the team will receive another notification of the Final Judgement.

Clarifications

To request a clarification on a problem, click on the **Clarifications** tab at the top of the page. This will display the **Clarifications** page, which will look something like this:

international collegiate programming contest

Elapsed: 00:42:10

TEAM 15

Runs

Clarifications

Options

Scoreboard

Logout

Remaining: 04:17:50

New Clarification

Filters

Recipient

- ☐ All Teams
- ☐ Just My Team
- ☒ Both

Select Problem ▼

Clear Filters

Recipient	Problem	Question	Answer	Time
team15	Hello	For the Dogwood problem, how tall can a tree grow?	There's no such problem in this contest.	43
team15	Sumit	For the Sumit problem, do we have to consider negative numbers in the input?	Answer Pending	42
All	Hello	For the Hello problem, does the output have to include the current time?	The answer to this question is given in the problem statement.	39

The Clarifications page displays all clarification requests previously submitted by your team. It also displays clarifications that the judges have chosen to send to all teams. If a clarification request has been answered, the answer also appears in the Clarifications table.



To request a **new** clarification, click on the **New Clarification** button in the upper left hand corner of the **Clarifications** page. A pop-up will display with the title “Submit a Clarification”, as shown here:

- Select the problem for which you are submitting a clarification request using the **Select Problem** drop down list.
- Enter your clarification request into the text box. *Only plain text is allowed.*
- Click the **Submit** button. A clarification must have a problem selected and text entered into the text box in order to be submitted.

Note that if you submit a clarification request which asks about something that the judges feel is adequately covered in the problem statement, they may simply respond to your clarification with an answer such as “No response; read the problem statement”.

- There are two **filters** that can be used to filter (limit) the clarifications being displayed in the clarifications table: *recipient* and *problem*.
 - Filtering by **Recipient**:
 - If you want to filter clarifications to view *only clarifications intended for All Teams*, click on the **All Teams** radio button under the Recipient filter heading.
 - If you want to filter clarifications to view *only clarifications intended for only your team*, click on the **Just My Team** radio button under the Recipient filter heading.
 - By default (if no recipient type filter is chosen) or if the **Both** radio button is clicked, all recipient types will be displayed.
 - Filtering by **Problem**:
 - If you want to filter clarifications to view *only clarifications submitted for a single problem*, click on the **Select Problem** drop down list. Select the problem you want to filter by.

Options

There are two WTI configuration options: changing your team login password and enabling/disabling certain pop-ups. *These options are available at the discretion of the contest administrator, and may or may not be available for your contest.*

To select available options, click on the **Options** tab at the top of the page. This will display the **Options** page, shown below:



To change your team login password (if enabled by the Contest Administrator):

- Click on the **Change Password** button located on the left side of the options screen.
- A pop-up will display with the title Change Password.
- Click on the **Current Password** box and enter your current team password.
- Click on the **New Password** box and enter the new password you want for your team login credentials.
- Click on the **Re-enter New Password** box and enter the new password you want again.
- Click on the **Submit** button.
- If you change your mind and decide not to change your password, click the **Cancel** button to be returned to the options screen.

To enable/disable certain pop-ups:

- To enable pop-ups, click on the **Enable Pop-ups** radio button.
- To disable pop-ups, click on the **Disable Pop-ups** radio button.

Logging Out

To log out of PC², click the **Logout** tab at the top of the page. This will return you to the **Login** page.

“About” Information

To view information about PC², click on the **About** link available at the bottom of any page. A pop-up will be displayed with information about PC² and its creators, similar to the following:

To exit from the “About” screen, click the **Close** button in the lower left-hand corner of the pop-up. This will return you to the page you were on prior to clicking the **About** link.



The PC² Command Line Interface

The PC² command line interface may be used to make submissions or retrieve your previous submission's source code.

Making Submissions

To submit a solution from the Linux command line, use the following command:

```
pc2submit <problem>.<extension>
```

where **<problem>** is the label of the problem (letter or short name) and **<extension>** is a standard extension for your language (eg. **c**, **cpp**, **py**, **java**, **kt**, etc.) If you follow this format for the submission's filename, then the problem and language will automatically be detected and you do not have to specify them on the command line. **pc2submit** will also try to auto-detect the main class (for Java and Kotlin) or the main file (for Python).

pc2submit also accepts a variety of arguments for explicitly specifying details about the submission. For example, if your **C++** solution to problem **E** is called **E.cpp**, you could use any of the following examples to submit your solution (there are others as well):

```
pc2submit E.cpp
pc2submit -l cpp E.cpp
pc2submit --language cpp E.cpp
pc2submit -l C++ -p E E.cpp
pc2submit --problem E E.cpp
```

You can override the auto-detection performed by **pc2submit**; for a complete list of all options and examples, type the command **pc2submit --help**.

On Windows systems, if Python is installed on your system then the **pc2submit** command can be used by invoking it with a command like:

```
python pc2submit E.cpp
```

(Continued on the next page)



Retrieving Your Previous Submission's Source Code

To see a list of your previous submissions, use the following command:

```
$ pc2submit -L
```

Sub#	Problem	Language	Time Submitted	Time Into Contest
8	K:SLA Tomography	C++	2025-06-17 10:04:50.290-04	00:16:44.331
69	B:Circle of Leaf	C++	2025-06-17 10:25:33.359-04	00:37:27.399
75	B:Circle of Leaf	C++	2025-06-17 10:27:10.193-04	00:39:04.233
127	B:Circle of Leaf	C++	2025-06-17 10:48:31.384-04	01:00:25.424
163	H:Ornaments on a Tree	C++	2025-06-17 11:02:20.139-04	01:14:14.180
227	A:A Totient Quotient	C++	2025-06-17 11:28:02.888-04	01:39:56.929
230	L:Solar Farm	C++	2025-06-17 11:29:09.308-04	01:41:03.348
252	L:Solar Farm	C++	2025-06-17 11:37:53.522-04	01:49:47.563
271	L:Solar Farm	C++	2025-06-17 11:45:28.022-04	01:57:22.061
354	D:Geometry Rush	C++	2025-06-17 12:17:36.740-04	02:29:30.779
465	D:Geometry Rush	C++	2025-06-17 13:00:34.801-04	03:12:28.841
473	F:Mob Grinder	C++	2025-06-17 13:03:28.080-04	03:15:22.118
543	F:Mob Grinder	C++	2025-06-17 13:33:47.282-04	03:45:41.324
552	B:Circle of Leaf	C++	2025-06-17 13:39:23.952-04	03:51:17.991
555	B:Circle of Leaf	C++	2025-06-17 13:39:53.221-04	03:51:47.262
572	F:Mob Grinder	C++	2025-06-17 13:45:01.327-04	03:56:55.368
575	F:Mob Grinder	C++	2025-06-17 13:46:34.895-04	03:58:28.934
621	B:Circle of Leaf	C++	2025-06-17 14:03:14.779-04	04:15:08.821
712	E:Humans vs AI	C++	2025-06-17 14:35:02.369-04	04:46:56.413
732	E:Humans vs AI	C++	2025-06-17 14:39:34.878-04	04:51:28.921

To retrieve the source code for one or more of your submissions, you can use the **-F** option of `pc2submit` along with one or more submission numbers (Sub# as shown in the example above):

```
pc2submit -F 69 271 465
```

That command would fetch the source code for **Sub#**'s 69, 271 and 465. Each will be placed in its own sub-folder with the name **sub_#**, eg. **sub_69**, **sub_271**, **sub_465**:

```
$ ls -l sub_*/*
-rw-r--r-- 1 team99 team99 1287 Jun 18 07:58 sub_271/L.cpp
-rw-r--r-- 1 team99 team99 4510 Jun 18 07:58 sub_465/D.cpp
-rw-r--r-- 1 team99 team99 1619 Jun 18 07:58 sub_69/B.cpp
```