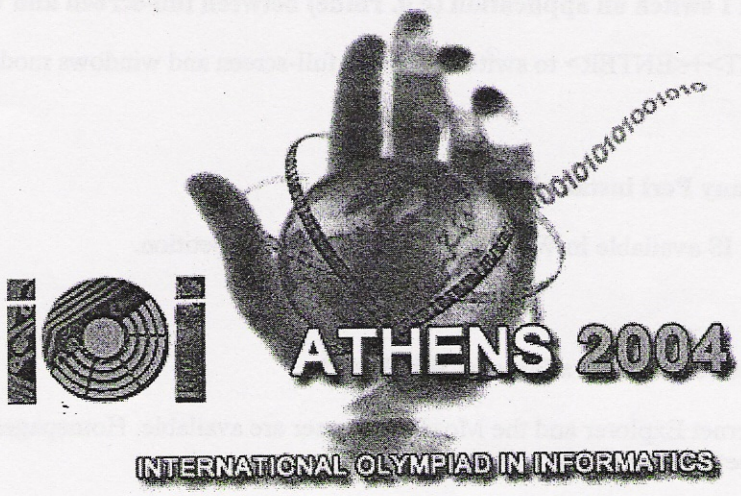
Competition Environment FAQ



The Windows XP Desktop

Q: What is the username/password to login?

A: Username: ioi, Password: ioi

Q: Where should I store my files?

A: Use C:\IOI folder and its subfolders. A shortkey to the C:\IOI folder exists in your desktop.

Q: What is the starting folder of programs I begin using the desktop shortkeys?

A: All applications (IDEs, editors, command prompt etc) start in the C:\IOI folder.

Q: How can I switch an application (e.g. rhide) between full-screen and windows mode?

A: Use <ALT>+<ENTER> to switch between full-screen and windows mode.

Q: Is there any Perl installed in Windows?

A: Yes, Perl IS available in Windows for this year's competition.

Q: What browsers are available in Windows?

A: Both Internet Explorer and the Mozilla browser are available. Homepages have been set to the competition server, http://www.ioilan.org

Q: How can I record a CD?

A: Insert blank CD and select "Open Writable CD Folder". Drag & drop the files you want, and select "Write these files to CD"

Q: Are there any ergonomic tools available?

A: An ergonomics aid named "RSIGuard" is installed; you may start it by clicking its icon on the desktop. It watches your typing activity and suggests appropriate times for you to take typing breaks.

Rhide in Windows

Q: Should I use rhide for my Pascal programs?

A: Rhide has many problems when dealing with Pascal programs. Use the Free Pascal IDE instead.

Q: Rhide crashes whenever I try to open an existing file from the editor.

A: First open a new blank file ("*File->New*"), then open the file you want (F3).

Q: The Rhide help (F1) doesn't work under Windows.

A: You can use the online Rhide help available on the competition server at http://www.ioilan.org/help/rhide/

Q: Rhide crashes when I use context-sensitive help (<CTRL>-F1) many times.

A: Do not use <CTRL>-F1 to get help. Use the standard C library reference on your desktop to get help on the C library. Use the Free Pascal IDE reference on the competition server to get help on Pascal commands. Use the STL reference on the competition server to get help on STL templates. See the "*Competion Server and the Grading System*" section below for URLs.

Q: If I set any breakpoints and my program terminates, rhide crashes.

A: It is the termination that causes the crash. Set a breakpoint at the end of your program, and when it is reached reset the program (<CTRL>-F2) instead of letting it terminate.

Q: The debugger doesn't work for Pascal programs.

A: Use the Free Pascal IDE, which has a very similar interface and works much better.

Q: Rhide doesn't seem to compile or link Pascal programs at all.

A: Name all your Pascal programs with the *.pp* extension (e.g. *hello.pp*). Create a project and add your source file to it, or set "*Project -> Main Target Name*" to the name of your source file without the extension (e.g. *hello*) and set "*Project -> Primary File*" to the full name of your source file (e.g. *hello.pp*).

Q: Rhide still doesn't compile/link Pascal programs! It doesn't run them either!

A: Rhide doesn't read its configuration file, *rhide.env*, unless it is in the folder rhide started from. Copy rhide.env to any folder you begin rhide from. It already exists in C:\IOI and C:\. You can also retrieve a copy at http://www.ioilan.org/files/rhide/windows/rhide.env

Q: I tried everything and rhide still doesn't work. What should I do?

A: Use the Free Pascal IDE, or the command-line tools (gcc,gdb etc)

The Redhat Linux Desktop

Q: What is the username/password to login?

A: Username: ioi, Password: ioi

Q: What desktops are available?

A: Both GNOME and KDE are available. Default is GNOME; select KDE by clicking on the "Session" button on the login screen.

Q: Where should I store my files?

A: Use the directories below /home/ioi.

Q: What browsers are available in Linux?

A: Mozilla is available. Homepage has been set to the competition server, http://www.ioilan.org

Q: How can I switch to a text-mode (full screen) console? How can I return to XWindows?

A: You can use <CTRL>-<ALT>-F1 to enter a console, <CTRL>-<ALT>-F7 to return to X.

Q: How can I use a floppy disk?

A: In GNOME right-click on the desktop and select *Disks -> Floppy*. An icon of a floppy will appear, double-click to open a window to the floppy. In KDE the icon is already on the desktop.

Q: How can I record a CD?

A: Double click on the "CD Recorder" icon to open K3b. Click the "New Data CD Project", and then drag & drop files into the DataCD1 window. Click on the "Burn" button, then click on "Burn" again. Multisession options are available in the "Settings" tab.

Q: Are there any ergonomic tools available?

A: An ergonomics aid named "Xwrits" is installed; you may start it by typing 'xwrits &' in a shell. It watches your typing activity and suggests appropriate times for you to take typing breaks. Use 'man xwrits' for detailed help on using xwrits.

Q: How can I configure a different keyboard?

A: Run 'redhat-config-keyboard' in a shell.

Q: How can I view PDF files?

A: Use either 'ggv' or 'xpdf'.

Rhide in Linux

Q: Rhide doesn't seem to compile/link Pascal programs at all.

A: Name all your Pascal programs with the *.pp* extension (e.g. *hello.pp*). Create a project and add your source file to it, or set "*Project -> Main Target Name*" to the name of your source file without the extension (e.g. *hello*) and set "*Project -> Primary File*" to the full name of your source file (e.g. *hello.pp*).

Q: Tried that, but rhide still doesn't link my programs!

A: Rhide doesn't read its configuration file, *rhide.env*, unless it exists in the folder rhide started from. You need to copy rhide.env to any holder you begin rhide from. It already exists in your home directory, /home/ioi. You can also retrieve a copy at http://www.ioilan.org/files/rhide/linux/rhide.env

Q: The UserScreen <CTRL>-F5 is not available when I'm running Rhide in XWindows.

A: You can run Rhide in a shell, and standard output will be directed to the terminal window. Or you can run Rhide in a full-screen console.

Q: I tried everything, but rhide still has problems.

A: Use one of the alternatives (the kate editor, the ddd debugger or the command-line tools).

Competition Server and Grading System

Q: What is the URL of the competition server?

A: The URL of the competition server is http://www.ioilan.org. It contains links to online manuals, files you might need to download, and a link to the grading system itself.

The homepage of your browsers has been configured to connect to the competition server when they start.

Q: What help files are available on the competition server?

A: The latest version of this document, check it out for updates: http://www.ioilan.org/help/manual.html

The competition rules: http://www.ioilan.org/help/rules.html

A description of the competition environment: http://www.ioilan.org/help/environment.html

Free Pascal manual: http://www.ioilan.org/help/fpc/

Rhide documentation: http://www.ioilan.org/help/rhide/

STL reference: http://www.ioilan.org/help/stl/

Q: How do I login to the grading system?

A: You can find the grading system at http://www.ioilan.org/ioigate.

Use your ID (i.e. USA03, GRE02 etc) and the password supplied to you before the beginning of the competition to enter the grading system. Use the "*Logout*" link at the bottom of the main contest page to logout.

Q: What can I do with the grading system?

A: You use the grading system to submit task solutions, submit programs and test data for custom runs, print programs, back up your files, and view your backups. You can also download previously saved files back to your computer, in case you accidentally delete them or you are moved to a different computer.

Q: How do I submit my programs for grading?

A: To submit a solution for a programming task, you must put a header at the top of your source file, so the grading system can determine the task being solved and the programming language being used. The headers are:

for C	for C++	for Pascal

`/*`
`TASK: task-name`
`LANG: C`
`*/`
	`/*`
	`TASK: task-name`
	`LANG: C++`
	`*/`
		`(*`
		`TASK: task-name`
		`LANG: Pascal`
		`*)`

Q: What should I do if I am asked to submit only the solution (output) of my program?

In case you are asked to submit only the solution (output) of your program, the first line should be:

`#FILE task-name case-number`

The task descriptions for file tasks will give complete examples of the output files.

Q: How can I submit a program?

A: Select a file to submit using the "*Browse*" button, then click on the "*Submit*" button.

Q: If I have submitted multiple versions of a program, which one is graded?

A: The most recent successful submission will be graded.

Q: How can I get back a submitted program?

A: Click on the "*See submitted solutions*" at the bottom of the page, and select a version you want retrieve.

The most recent successful submission can also be downloaded from the main page of the grading system.

Q: How can I store a file for safekeeping? (backup)

A: Select a file to submit using the "*Browse*" button, then click on the "*Backup*" button.

WARNING: Files you submit for safekeeping will **NOT** be graded. Only programs you have submitted using the "*Submit*" button will be graded.

Q: How can I get back a backup file?

A: Click on the "*Restore*" button, and select the file you want to retrieve.

Q: How can I print a file from my disk?

A: To print a file from your disk, select it using the "*Browse*" button, then click on the "*Print*" button. Your printout will be delivered to you as soon as possible. There is a limit of approximately 10 pages printout per file; if your file is larger than that, only the first 10 pages will be printed.

Q: How can I run my program on the server, using my own test data? How can I see how well my program runs when the server is enforcing the time and memory limits?

A: Select your program and a file with your own test data using the "*Browse*" buttons just bellow the green area of the main contest page, and click on the "*Test*" button. Your program will run on the server, using your own test data as input, but the server compiler flags, time and memory limits will be used. Any program you test this way will **NOT** be saved by the grading system. Again, only programs you have submitted using the "*Submit*" button will be graded.

Q: Why should I check how well my program runs on the server, using the "Test" facility? Should I bother if I'm already programming in Linux?

A: It is important to note that the grading system runs on Linux (in an environment identical to the provided Linux installation), and it is only the Linux behavior and performance which will be used to assign points. Windows users may use the "*Test*" facility of the grading system to evaluate the behavior and performance of their programs under Linux. Even for Linux users, since the grading system enforces memory limits on program execution, output sizes, etc., contestants may benefit from using the Test facility.

Compilation and Execution Environment

Q: What is the server's compilation commands?

A: For C: `gcc -pipe -O2 -o task task.c -lm extimec.o`

For C++: `g++ -pipe -include /usr/include/stdlib.h -O2 -o task task.cc -lm extimecp.o`

For Pascal: `ppc386 -So -O1 -XS -otask -kexectime.o task.p`

Q: What is the extra object file (extimec.o, extimecp.o, exectime.o) mentioned in the compilation commands?

A: This object file provides a function `exectime` that you can call to discover how long your program has been running. The function declarations necessary to use them are:

C: `int exectime( void )`

C++: `int exectime( )`

Pascal: `function exectime: longint; cdecl; external;`

In all cases, exectime returns the number of milliseconds that the program has used.

Q: Where can I find the exectime object files?

A: In Windows, they are in C:\exectime.

In Linux, they are in /usr/local/exectime/

You can also download the object files from http://www.ioilan.org/files/exectime/

Copy these files to the same directory as your program in order to link them using the commands described above.

Q: What exit code should my program use when it terminates?

A: Your program must exit with exit code 0. In Pascal, this is the default: no special steps are necessary. To exit with code 0 from C or C++, `return 0;` from inside `main`, or call `exit(0);` to exit the program. Note that to use `exit()` you must `#include <stdlib.h>`.

Q: What are the integer sizes recognized by the compilers?

A: The compilers provided in this competition recognize the following signed integer types:

	8 bits	16 bits	32 bits	64 bits
C/C++	char	short	long	long long
Pascal	Shortint	Integer	Longint	Int64

Note that an `int` in C is 32 bits, while an `Integer` in Pascal is 16 bits.

Q: How should I use random seeding? Can my programs be non-deterministic?

A: Your programs are not allowed to be non-deterministic: they must produce the same output each time they are run on a particular input. If you want to use a pseudo-random number generator, you can get deterministic behavior by seeding it with a fixed value. For example, in C and C++ you can use `srand(0);` and then call `rand()` to get a fixed sequence of random values. In Pascal, you can set `RandSeed:=0;` and then use `Random()` to get a fixed sequence of random values.

Q: My C++ program doesn't work when I'm using templates or the STL.

A: One difference between `g++` v3 and Borland C++ is that `g++` now enforces the C++ namespace conventions, which place "standard library" symbols in the '`std`' namespace. Practically, this means that you must either qualify all these symbols with '`std::`' (e.g. '`std::cin`') or place the declaration '`using namespace std;`' under your '`#include`' lines, e.g.:

```
#include <iostream>
using namespace std;
```

Borland C++ requires neither of these steps.

Q: How repeatable is task timing?

A: 0.01 seconds.