

# CeeBot

## CeeBot4

USER HANDBOOK



an EPSITEC product

[info@ceebot.com](mailto:info@ceebot.com)

[www.ceebot.com](http://www.ceebot.com) [www.epsitec.ch](http://www.epsitec.ch)



# 1 Introduction

## 1.1 *Minimal requirements*

CeeBot requires a reasonably recent computer. It is particularly important to have a good 3D graphic adapter for maximum performance.

- 500 MHz CPU
- 3D graphic card with 16 Mb RAM
- 200 Mb free disk space
- Windows Vista, XP, 2000, Me, 98
- DirectX 8 (If necessary, the full version of CeeBot will install DirectX 8)

However CeeBot4 will probably work reasonably well even on low-end onboard graphic chips. Some graphic chips in notebooks will not provide satisfactory performance.

## 1.2 *To install CeeBot from a CD*

- Insert the CeeBot CD-ROM in the drive

It should start automatically after ten seconds or so.

Click **Install**

If the autorun doesn't start, follow these steps:

- Double-click **My computer**
- Open the **D:** drive (where D: is the letter of your CD-ROM drive)
- Double-click **Install**

The installation process will check if you have DirectX® 8a. If this is not the case, you'll be prompted to install it.

If you have a previous version of CeeBot, a new installation in the same folder might overwrite all data, saved games or programs (depending on where this data is stored).

## 1.3 *To download CeeBot from the Internet*

If you download a free version from our website, a downloader program will take care of everything. It will download the files safely, restarting where it stopped in case of a problem and create a folder with the installation files. You can burn this folder into a CD in order to keep it for later use.

If you ordered the SCHOOL or COLLEGE version of CeeBot, you will get an email with links to download the necessary files. These links point to a downloader program which will download the files safely and create the three versions, i.e. the TEACHER, CAMPUS and STUDENT versions, each in one folder.

The same email will contain the license files which are needed to execute the program. Just copy each license file into the **data\** folder in the respective installation files folder.

You should make CDs of the programs for backup. The files might not remain for ever on our server. Just burn the content of the three installation files folders on one CD each. As soon as you insert one of these CDs into the drive, the autorun should start automatically.

## ***1.4 Configuring CeeBot4 for teaching***

Chapter 7 gives more information about how to install CeeBot on a local network.

## ***1.5 Execute CeeBot4 or show the readme file***

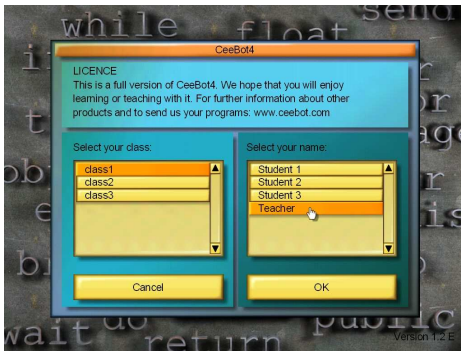
After you installed CeeBot4 on your computer, click "Start". The "Programs" menu contains a new folder "CeeBot4". You will find there a shortcut to CeeBot4 and to this readme file, which contains valuable information about how to use CeeBot4.

## ***1.6 To uninstall CeeBot***

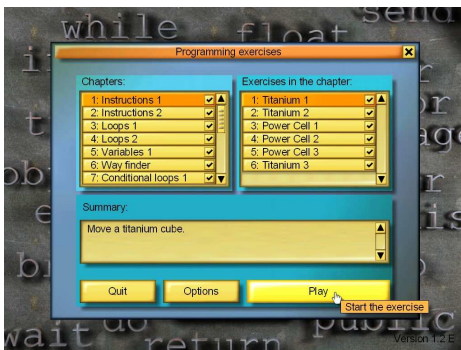
- Double-click **My computer**.
- Double-click **Control panel**.
- Double-click **Add or Remove programs**.
- Double-click **CeeBot4** in the list.

## 2 First approach

The detailed explanations will be found in the following chapters. This is your very first contact, to help you get familiar with CeeBot.



The first thing CeeBot needs is the player's name. Each player must use a separate name, because CeeBot saves your programs and keeps track of your progress. If you have got a CAMPUS version, choose the user name **Teacher**, so that you can access the settings and the solutions for every exercise. In the DEMO, PRIVATE and TEACHER versions, solutions and settings are always available, in a STUDENT version, only the settings are available. Then click **OK**.



CeeBot now lists the various exercises. The left column shows the different chapters, and the right column is the list of the exercises in the selected chapter.

In some versions, you can choose the exercise series with three or four options on top the window. Choose the **Standard** series.

The exercises get progressively more difficult. We suggest you try them in this order, even though this isn't compulsory.

Choose **1: Instructions 1** and **1: Titanium 1**, then click **Play**.



The green text at the top of the screen asks you to press **F1** to get the instructions for the current exercise.

You can also click this button, at the bottom of the screen:





The information displayed is the same as in these pages. To close this window, press the button in the bottom left corner.



If necessary, you can come back to this screen at any time by pressing **F1**.



Click the robot in order to select it.



In the lower left corner of the screen select the first program in the list. Each robot can have up to 10 programs. This can be handy when you're testing different solutions. All these programs are saved automatically in each player's profile.



Program number 10 (use the scroll bar) contains one possible **Solution** for the exercise (depending on your version and the user name you chose). You can immediately execute it with the following button:



To enter the program editor, click **{..}**. If you selected program 10, you will see the solution.



If you selected an empty program, the editor comes up with an empty shell ready for you to write your new program.

The text cursor is between the braces.

Do not alter the first line or the braces.





Program the robot so that it drops the titanium cube on the plate-form. In order to achieve this, it must grab the cube, move 5 meters forward, and drop the cube. Enter the following instructions:

```
grab();
move(5);
drop();
```

Click **OK**. If you've made a mistake, it will be highlighted in blue, and a message will be displayed at the bottom of the screen.

If everything is correct, when you click **OK** the editor is closed.



Click this button to execute the program. The robot should now grab what is in front of it, move 5 m forward, and drop the titanium cube. The exercise will finish automatically, and return to the main menu.

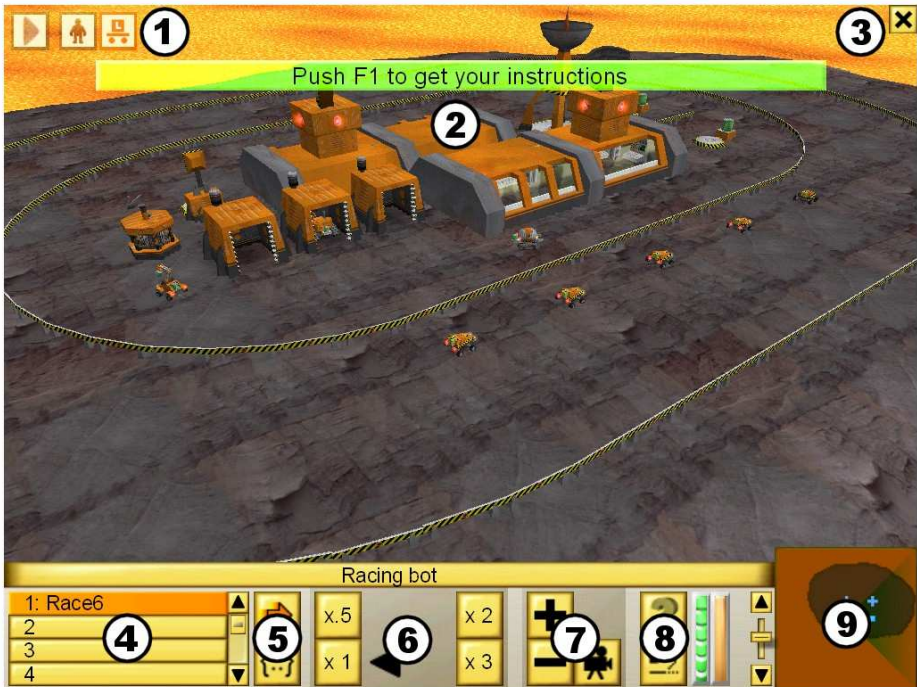
That's it; you just did the first exercise.



You can now take a look at the following exercises. The students will progressively learn various control structures which are to be applied in more and more complex environments. The technically most challenging exercises are doubtlessly the robot soccer exercises, where the students use vector mathematics and classes in order to stage well-coordinated soccer matches. If you want some action, you should have a look at the last exercises, staging fights against giant insects and other robot teams.

### 3 The screen

During an exercise, the screen looks as follows:



- 1 Shortcuts to the robots and the astronaut
- 2 Temporary messages
- 3 Access to the menu
- 4 List of the programs
- 5 Programming controls
- 6 Execution speed controls
- 7 Camera controls
- 8 Access to the help texts and exercise reset
- 9 Mini-map (provides an overview of the scene)

#### 3.1 Shortcuts to the robots and the astronaut

The shortcuts on the top of the screen allow you to easily select the different robots in the scene, even if they are out of sight.

#### 3.2 Temporary messages

Temporary messages appear on the upper side of the screen, either messages to guide the students, or error messages during the execution of a program.

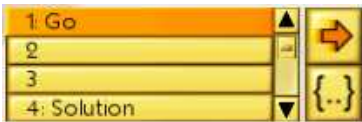
### 3.3 The menu



If you click on the cross in the top right corner of the screen, the following menu appears:

<b>Continue</b>	Closes the menu in order to continue the exercise.
<b>Options</b>	Access to the options. Some options are not accessible. In order to get access to all the options, you must close the exercise, and click on <b>Options</b> in the main menu (see section 8). In the CAMPUS version, the options are only available with the username <b>Teacher</b> .
<b>Restart</b>	Restarts the exercise from the beginning. The programs you have already written will not be deleted.
<b>Close</b>	Closes the current exercise and displays the main menu.

### 3.4 The list of the programs



With the list on the lower left corner you can access the 10 programs in the robot. The first four rows display the names of the first four programs. Use the scroll bar to access the following six programs. The last programs contain one or several **Solutions** (in the version CAMPUS only if you selected the user name **Teacher**).

### 3.5 Programming controls



This button executes or stops the selected program.



This button opens the editor for the selected program (see section 5.1). You can also execute a program while the editor is open, so that you can see more easily how a program works or why it does not work.

### 3.6 Execution speed controls

The students should spend their time programming and not waiting for the robot to finish its task. However, a slow execution speed allows to check the correct functioning of the program. Therefore a few buttons allow you to set the execution speed of the program.



Click this button for half execution speed. This can be useful if events are unfolding too fast.



Click this button for normal execution speed.



With these buttons you can set the speed to twice or three times faster.





### 3.7 Camera controls



With this button you can change the camera viewpoint. It switches between a view from above and a view from behind.



Increases the distance between the selected robot and the camera.



Reduces the distance between the selected robot and the camera.

### 3.8 Access to the help texts and reset



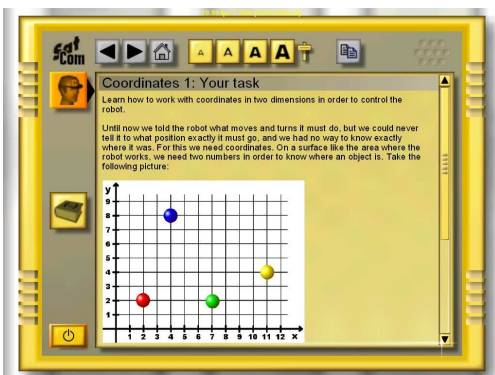
A click on this button will open a window that displays the instructions for the current exercise (see the following section).



This button will reset the exercise. When the program did not do what you expected, use this button to put the robots and other objects in the scene back to their initial place.

## 4 Displaying instructions and help texts

A special window is used in CeeBot in order to display the instructions for the current exercise and other help texts about the CBOT programming language.



It works like an internet browser. Every time a word is underlined in blue, you can click it in order to access a different page about this subject. At the end of the instructions for every exercise there are some links to pages with additional information.



Shows the previous page



Shows the following page



Shows the home page, which appeared when you opened the window.



Copies the selected characters into the clipboard. Like this you can copy a program displayed here into the program editor (see section 5.1).



Shows the instructions for the current exercise. The key **F1** opens this page directly.



Displays the page with the CBOT reference home page. The key **F2** opens this page directly.



Closes the window and returns to the exercise.

When one of the buttons is grey, this function can not be used.

## 5 Programming

The goal of CeeBot4 is precisely to learn how to program various kinds of robots in various environments.

### 5.1 Program editor

To enter the program editor proceed as follows:

- Select a robot
- Choose one of the 10 program slots in the bottom left corner of the screen
- Click the **Edit the selected program {..}** button



The editor works like any other Windows editor. The keywords of the CBOT programming language are displayed on a colored background, so that you can easily check the syntax. The automatic indentation according to the braces shows the structure of the program.

If you open the editor while a program is running, you will get information about how it is executed (see section 5.1.11).

When the cursor is on a keyword, the status bar at the bottom of the program editor window displays the word and some additional information. You can then click on the status bar or press **F3** to get further information.

You can double click a word to select the whole word. **Shift-arrow** also selects text. **Ctrl-arrow** moves word by word and **Shift-Ctrl-arrow** selects text word by word just like any standard editor. The usual **Ctrl-X**, **Ctrl-V**, **Ctrl-C** shortcuts for copying and pasting text are also available.

### 5.1.1 New



Clears the whole program and creates an empty skeleton program:

```
extern void object::Move1( )
{
}

```

`Move1` is the default name of the program attributed by CeeBot4 depending on the current exercise. You can change it to whatever you like, but you mustn't use blank spaces or special characters, only letters and digits.

**Note:** You can revert to the previous program by pressing **Ctrl-Z** or by clicking the Undo button (see below).

### 5.1.2 Open and save



All programs you create are automatically saved within the exercise. On the other hand, if you want to reuse a program in another exercise you can save it by hand.

With the following four checkboxes at the bottom of the dialog you can choose the folder where you want to save or read:

- Private**      The program is saved in a private folder under the selected class and student name.
- Public**        The program is saved in a public folder and will be available to all other students.
- Files**          The program is saved in the files folder, where it can be read with the files management instructions of the CBOT language. This can be useful if you want to write a program into a robot when it is created by another robot.
- Backup**        When you program a robot which was created during the mission, you will not find it again in this robot if you close and reopen the exercise, or if the robot is destroyed in a fight. Such programs will be saved automatically in the **backup** folder, where you can find them again.

The name of the folder where the program will be saved appears on top of the **Save** dialog. For example, **Savegame\Class\Student\Program\** means that the program will be saved in **C:\Program Files\CeeBot4\Savegame\Class\Student\Program\** (assuming CeeBot has been installed in the folder **C:\Program Files\ CeeBot4\**). This

allows you to access the program from another application, for example to send a program by email to another school.

**Shortcuts: Ctrl+O and Ctrl+S.**

### 5.1.3 *Undo*



Cancels the last modification in the program. You can Undo the 20 last modifications.

Shortcut: **Ctrl+Z**.

### 5.1.4 *Cut, Copy and Paste*



*Cuts* or *copies* the selected text to the clipboard. If no text is selected, the whole line will be taken. The contents of the clipboard can be pasted into the program you are editing.

Shortcuts: **Ctrl+X**, **Ctrl+C** and **Ctrl+V**.

### 5.1.5 *Font size*

Use this slider to change the font size for the editor and the information display window.

### 5.1.6 *Instructions*



Shows information about the goal of the current exercise. Shortcut: **F1**.

### 5.1.7 *Programming help*



Displays the homepage of the CBOT reference text, or a help specific to the chapter (chapters 1-5). Shortcut: **F2**.

### 5.1.8 *OK*

Compiles the program and quits the editor. If there is a syntax error in your program, the error is highlighted and an appropriate error message is displayed in the status bar.

### 5.1.9 *Cancel*

Quits the editor without compiling but all modifications are saved.

### 5.1.10 *Compile*



Compiles the program without quitting the editor. This is useful to check your program for syntax errors.

### 5.1.11 Execute/stop



Starts or stops program execution without closing the editor. This is useful for debugging purposes as you can follow the progress of your program.

If the button on the right side (see below) shows a little standing man, the program will be executed step by step.

**Note:** During program execution the content is displayed in orange and you cannot modify the program.

### 5.1.12 Pause/continue



Switches from step by step execution to continuous execution and back.

### 5.1.13 One step



Executes the next instruction in step by step mode. The lower part of the program editor shows the contents of the different variables which change during the progress of the execution.

## 5.2 The CBOT language

The CeeBot programming language is very close to Java™ and C#. It has been designed specially for CeeBot and is very well adapted to teaching purposes.

## 6 Using CeeBot4 in a classroom

The best way to get an idea about how CeeBot4 can be used in a classroom is to take a look at one exercise after another. Depending on version and user name, you will find one possible solution to all the exercises in the program slot 10. Here are some further considerations that could be useful to you.

### 6.1 CeeBot and the cup of coffee

All the products of the CeeBot series are made so that the teacher can install the program on all computers, let the students sit down in front of the computer and go to drink a cup of coffee. However, this is not exactly the use that we recommend.

In accordance with modern pedagogical practice, we try to reduce the time that the teacher spends standing in front of the class trying to explain whatever he or she is trying to teach them. It is much more motivating for both teachers and students alike to let every student progress at his or her own pace. CeeBot4 is perfectly adapted to this approach.

Just let the students work individually or in small groups. The teacher's role is to assist the students in their work. They might encounter practical difficulties like syntactical or algorithmic problems. Therefore the teacher must be familiar with the different concepts taught in the various chapters. In the exercises that leave more freedom to the students, the teacher has also a more coach-like role: often the students have very ambitious projects, and the teacher's role is to help them to set up a realistic project and to give them some pointers toward concrete solutions.



## 7 Installing CeeBot in a classroom

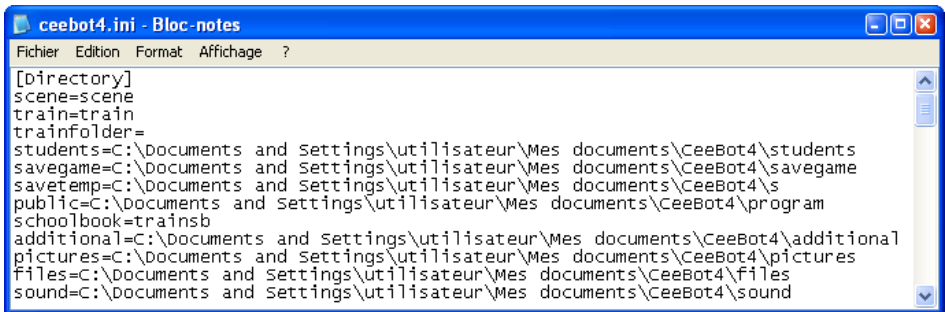
In order to install CeeBot in a computer classroom, you need to perform a few extra actions in order to enter the names of the students of the different classes and possibly to set up its use on a LAN.

### 7.1 *Setting up the folders*

The first time you launch CeeBot4 after installation with the standard installation files, a file **ceebot4.ini** will be created in the folder **My Documents\CeeBot4**. This file contains all the settings, among other the folders where CeeBot4 will find and write all the information.

In the versions CAMPUS and TEACHER (but not CAMPUSCD) you can influence the location of this file. When you launch CeeBot4, the program will look for this file first in the folder where the program was installed (usually **Program Files\CeeBot4**). If this file does not exist, it will try to read the file **ceebotpath.ini** in this same folder. This file must contain one single line with the filename where CeeBot should look for all the settings, i.e. **Z:\CeeBot4\ceebot4.ini**. If no file **ceebotpath.ini** is found, it will look for the settings in **My Documents\CeeBot4\ceebot4.ini**. All these possibilities should make it quite easy to install CeeBot in a computer room.

The first lines of the settings file are especially important. They contain all the settings for the folders where the program will read and write its files:



```
[Directory]
scene=scene
train=train
trainfolder=
students=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\students
savegame=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\savegame
savetemp=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\s
public=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\program
schoolbook=trainsb
additional=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\additional
pictures=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\pictures
files=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\files
sound=C:\Documents and Settings\utilisateur\Mes documents\CeeBot4\sound
```

In the versions CAMPUS and TEACHER (but not CAMPUSCD) you can enter here another folder path, on the same computer or on a server. Under Windows XP or Vista in user mode, the programs can not write into the folder **\Program Files\**. Therefore the folders where CeeBot must write information are set by default to **My Documents\CeeBot4\...**

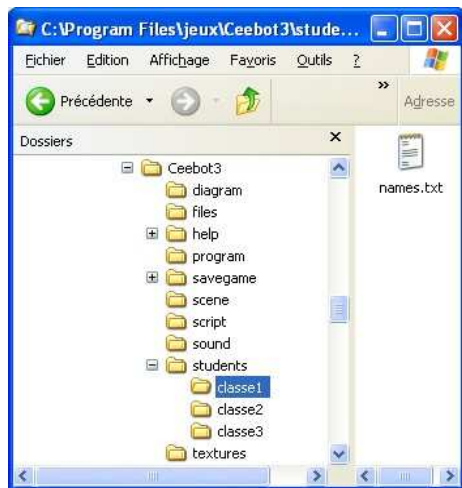
The folders defined with **scene=**, **pictures=**, **savetemp=** and **sound=** are not used in CeeBot4. The folder **savegame=** contains the programs made by the students, which are saved automatically. It also contains the programs saved into the private programs folder and the information about which exercises have been achieved successfully. The folder **public=** contains the programs saved with the button **Save** into the public folder. The folder **files=** is only used by the file management instructions, which are used in the missions to write programs into newly built robots.

The folder **train=** contains the standard exercise series, which is used if no specific exercise series is defined for a class or a student (see below). An exercise series

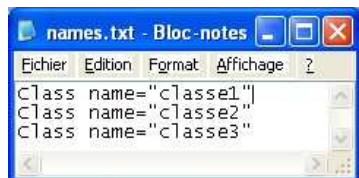
consists of a folder with several folders named **unitnn** where **nn** is a number ranging from 01 to 99. Each of these folders contains one or several folders **exnn** where **nn** is again a number ranging from 01 to 99. Each of these folders contains all the information necessary for one exercise. In list of chapter folders and exercise folders, the numbers must start with 01 and follow each other. As soon as there is a hole in the series, displaying of the chapters or exercises stops there. The folder **trainfolder=** can contain several folders with exercise series. This allows you to design specific exercise series for each class, or to use additional series. In some versions, the folder **schoolbook=** contains an additional exercise series which is specific to the level of the students targeted by this version.

The folder **additional=** can contain additional exercises downloaded from Internet or developed by the teacher. Another series of folders is more specifically designated for user exercises, you will find them within the **savegame=** folder in the folder **\\[class name]\\[user name]\\user**, their path can not be changed. The folders with these two exercise series work according to a different system than the normal exercise series. The folders containing the different chapters can have any name, they will be alphabetically sorted for display in the menu. The exercise folders within each chapter must have the same kind of name as in the standard folders, i.e. **unitnn** where **nn** is a number ranging from 01 to 99.

The folder **students=** contains one folder for every class. After the installation is completed, it contains for example the three folders **class1**, **class2** and **class3**:



Each of these folders contains a file called **names.txt**. The folder **\students\** itself also contains such a file. If you open the file **\students\names.txt**, you will find there the list of all the classes:



The file **names.txt** in each of the corresponding folders contains the names of the different students and other information like the **train** folder which should be used and the password (can be left empty). You will find a similar structure in the folder **savegame=**.

You do not have to work in these files in order to enter the student names but you should know about this structure in case you have to move the folders onto a server, or in case of a problem.

The folder **savegame=** allows for a more complex syntax than the other folders. If you specify simply one folder here, the files of the students will be saved in arborescence with one folder per class, and in each one of these on folder per student. Some schools have got already another structure for the individual files of their students. Therefore you can define a specific location for the class and student name in the path. **#c** stands for the name of the class, **#n** stands for the name of the student:

**savegame=\\server\\students\\ceebot** will save the files for each student in the folder **\\server\\students\\ceebot\\class1\\student1** (depending on the name of the class and student).

**savegame=\\server\\students\\#c\\#n\\ceebot** will save the files into **\\server\\students\\class1\\student1\\ceebot** (depending on the name of the class and student).

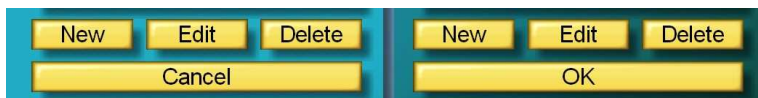
You can leave out one of the parameters, i.e. the name of the class if all students have got different names. As soon as there is one of the parameters in the path, the class and student names are not added any more at the end of the path.

## 7.2 Entering the students' names

In the STUDENT version, the number of classes is limited to 1, the number of students to 1. In the PRIVATE version, the number of classes is limited to 1, the number of students to 5.

Entering or changing the class or student names requires a "cheat code", so that the students can not do it. The code for the DEMO, PRIVATE, TEACHER and STUDENT versions is **studentsetup**. For obvious reasons, the code for the SCHOOL version will not be given here. You will get it in the email with the download links. However, we all know that such codes spread quickly among the students. Therefore we recommend setting the folder used for the student names (**students=**) to read-only access through the network, so that its content can only be modified directly from the server.

In order to enter the student setup mode, you must be in the menu where you can choose the user name. Push **Ctrl+Pause**, type the cheat code and hit the **Enter** key (works only with version 1.3 and later). A series of additional buttons appear at the bottom of the screen:



The button **New** allows you to enter a new class (button on the left side) or a new student (button on the right side). The class and student names are used as folder names, therefore the syntax of the names must correspond to the file name syntax rules (characters `\ / : * ? " < > |` are not allowed).

You can enter the name of a folder corresponding to a specific exercise series, in this case this name is added to the path defined with **trainfolder=**. If left blank, the train series will be taken in the folder defined with **train=**.

For students you can enter a password. Passwords can only contain the characters **0..9**, **A..Z** and **a..z**. Passwords are case sensitive. They are recorded in a coded form. However, the coding is not really state-of-the-art.

With the buttons **Edit** you can change the properties of a class or a student. However, this will work properly only if both **student=** and **savegame=** paths point to a centralized folder on a server. If you change the name of a class or a student, the corresponding folder in the **savegame=** folder will be renamed too, otherwise the students would loose the work they have done.

With the buttons **Del** you can delete a class or a student. A dialog asks you to confirm your intention. For security reasons, the corresponding folders (in the **students=** and **savegame=** folders) will not be deleted, but renamed, with the date and time of the action preceding the former name. In case of bad manipulation this allows you to restore the content.

### ***7.3 Installing CeeBot in a computer room***

There are a lot of setting to do after installing CeeBot before the students can actually use it. If you had to do these settings in every single computer, you would loose a lot of time. There is another way to do this. Actually the information written into the registry by the installation program is not essential. Once you installed CeeBot on one computer, you can therefore simply copy the folder **Program Files\CeeBot4** from one computer to the next and create a shortcut on the desktop.

You should of course either put the file **ceebot4.ini** into this same folder, or use a file **ceebotpath.ini** in order to put the settings file onto the server.

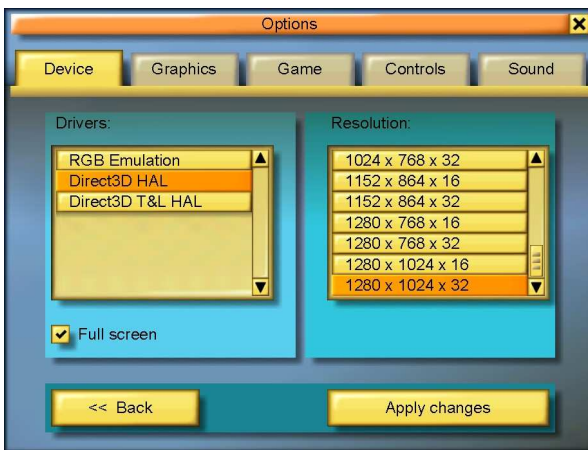
All the folder which contain the student names (**student=...**), the programs of the students which are saved automatically (**savegame=...**) and the public program folder (**public=...**) should be moved to a server. This will make your life much easier (only versions CAMPUS and TEACHER but not CAMPUSCD).

## **8 Options**

Click **Options** to access various settings, separated into 5 categories. This button appears in the CAMPUS version only if you have selected the user name **Teacher**. We mention here only the most useful settings. For the other options, a tool tip will give you further information.

### ***8.1 Display***

The first time you run CeeBot, it is set to use a 640 x 480 x 16 display. On many computers, you can get a much better graphic result by changing the default settings.



### Drivers:

It's best to choose a HAL driver (Hardware Abstraction Layer), and to avoid "Emulation" or "T&L" drivers.

### Resolution:

The first and the second figure stand for the size of the image on your screen. The third number represents the number of colors used:

16 bits will display 65,000 colors, whereas 32 bits will display 4 million colors. The higher you set these parameters, the more detailed the image will be on the screen. However, high parameters need much more processor and graphic power, and the frame-rate will be lower.

To find the best settings for your computer, we suggest you start off with 640 x 480 x 16 and climb slowly higher. Most modern graphic cards support at least 1024 x 768 x 16.

If the display seems to be a little jerky, try a lower quality.

### ✓ Full screen

Normally CeeBot runs in full screen mode, whatever resolution you choose. If you deactivate this flag, CeeBot will be displayed in a fixed size window of 640 x 480 pixels.

### [ Apply changes ]

Click this button to use the new settings.

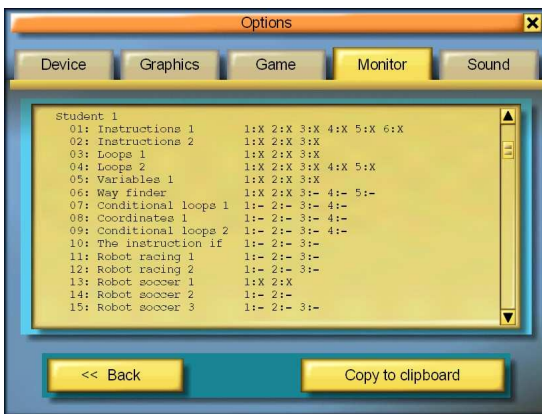
## 8.2 Graphics

If the game is too slow, or if the frame-rate is too low, you can tune down some of the graphic options. Obviously, the higher the settings the more beautiful the game!

## 8.3 Monitor

The monitor allows you to check your students' progress. It will check all the players in the **students** folder (see section 7.1) on this computer and show which exercises have been completed (on a network this may take a few minutes):





Therefore, if the folder **savegame=** is common to all computers and if all computers contain the same classes and students, it can sum up the progress of all students at once. You can copy the text to the clipboard in order to add it to your class record.

## 8.4 CPU usage on laptops

The CeeBot 3D-engine makes intensive use of the CPU and 3D card. This might cause overheating problems, especially on laptops. Experience has shown that running CPU-intensive programs on a regular basis for years on end might result in dust accumulation which prevents cooling and can result in hardware damage. This is due to fan activity which sucks dust into the computer where it gets stuck and prevents air flow.

We recommend opening and cleaning the inside of each laptop in intensive use every year in Spring, before the Summer period of greatest danger.

The CeeBot 3D engine was developed in 2000-2001; therefore most recent computers have got more than enough CPU power to run it. On laptops we recommend reducing the CPU use and therefore the frame rate in order to reduce energy consumption and dust accumulation. In the Graphics tab you will find the necessary settings.

## 9 Violence in CeeBot

CeeBot features several exercises which stage fighting against giant insects or between competing robot teams. This might seem a controversial feature. In the first versions, unlike previous programs based on the same 3D-engine like Colobot or CeeBot-A, there was no violence at all. However, several teachers asked us to put them back, because the students liked them.

We are of course concerned about growing violence in the world. We would like to offer an entirely non-violent, attractive learning environment. We tried, and failed: despite all our efforts, the non-violent version of CeeBot4 was considered by teachers who knew the previous programs to be much less attractive.

On the other hand, we think that motivating the students and therefore contributing to successful school careers will reduce violence more than if we stuck dogmatically to non-violent training material, when the media show real violence every day.

## 10 Development team

Otto Kölbl, Daniel Roux, Denis Dumoulin, Michael Walz, Martin Huber, Brian Ward, Michael Rundel, Timothy Loayza.

EPSITEC SA  
Rue d'Orbe 28  
CH-1400 Yverdon-les-Bains  
Switzerland

[info@ceebot.org](mailto:info@ceebot.org)  
[www.ceebot.org](http://www.ceebot.org)  
[www.epsitec.ch](http://www.epsitec.ch)