

# CeeBot

## CeeBot3 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.

- 300 MHz CPU, 64 Mb RAM
- 3D graphic card with 16 Mb RAM
- Sound card and loudspeakers or earphones
- 200 Mb free disk space
- Windows XP, 2000, ME, 98 or 95
- DirectX 8 (If necessary, CeeBot will install DirectX 8)

Recommended graphic cards :

- nVidia GeForce 2, 3 or 4
- Matrox G400, G450, G550 or Parhelia

However CeeBot3 will probably work reasonably well even on low-end onboard graphic chips. 3dfx Voodoo graphic adapters are not officially supported, but they may work (no guarantee). Some graphic chips in notebooks will not provide satisfactory performance.

## 1.2 To install CeeBot

- 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.

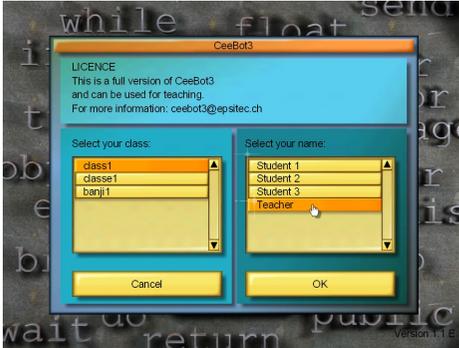
If you have a previous version of CeeBot, a new installation in the same folder will overwrite all data, saved games or programs.

## 1.3 To uninstall CeeBot

- Double-click **My computer**.
- Double-click **Control panel**.
- Double-click **Add or Remove programs**.
- Double-click **CeeBot3** 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. Choose the user name **Teacher**, so that you can access the settings and the solutions for every exercise. 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.

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

Choose **1: Move !** and **1: Move 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**.



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. 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.



The exercise consists in tracing a 5 cm-long line on the ground. First use the following instruction to lower the pen:

```
pendown() ;
```

Then move the robot 5 cm forward with the following instruction:

```
move(5) ;
```

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 lower the pen (it becomes transparent so that you can see what it draws), then move 5 cm forward, leaving a black trace behind.

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

In order to leave the exercise, click the cross button on the upper left side of the screen (or push the **Esc** key), then click **Close**.



You can now take a look at the following exercises. There are more than 100 exercises, starting with the instructions to control the robots. Then the students will progressively learn to program animations with hundreds of frames and a sound track. They will finally apply all the skills they have acquired in the final group or class project, where every student can contribute in the domain he or she likes most.

### 3 The screen

During an exercise, the screen looks as follows :



- 1 Temporary messages
- 2 Access to the menu
- 3 List of the programs
- 4 Programming controls
- 5 Execution speed controls
- 6 Camera and reset
- 7 Access to the help texts

#### 3.1 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 the programs.

### 3.2 The menu



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

- Continue** Closes the menu in order to continue the exercise.
- Options** 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 **Options** in the main menu (see section 8).
- Restart** Restarts the exercise from the beginning. The programs you have already wrote will not be deleted.
- Close** Closes the current exercise and displays the main menu.

### 3.3 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 (number 10 and before) contain one or several **Solutions** if you selected the user name **Teacher**.

### 3.4 Programming controls



This button executes or stops the selected program.



This button opens the editor for the selected program (see section **Erreur ! Source du renvoi introuvable.**). 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.

### 3.5 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 normal execution speed.



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



Click on this button in order to activate the **flash** mode: all movements of the robot will then be executed immediately, so that the drawing will be ready in less than one second.

### 3.6 Camera and reset



With this button you can change the camera viewpoint. One click changes to the top view where you can see the whole drawing surface, one more click comes back to the normal mode where the camera follows the robot. You can also just hit the space bar.



This button will reset the exercise. When the program did not do what you expected, use this button to clear the drawing and to put the robot back to its initial place.

### 3.7 Access to the help texts



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

## 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 **Erreur ! Source du renvoi introuvable.**).



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



Displays the page with the programming instructions that you need to do the exercise, with a short description. 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 CeeBot3 is precisely to learn how to program the robot, which can be found at the beginning of each exercise at the bottom left corner of the drawing surface.

### 5.1 Program editor

To enter the program editor proceed as follows:

- 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 5.1.12).

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 CeeBot3 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 must save it explicitly.

**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.

The name of the folder where the program will be saved appears below the title bar of the **Save** dialog. For example, **Savegame\Class\Student\Program\** means that the program will be saved in

**C:\Program Files\CeeBot3\Savegame\Class\Student\Program\** (assuming CeeBot has been installed in the folder **C:\Program Files\CeeBot3\**). 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 *Save picture*



After the robot finished drawing the picture, you can save it with a click on this button. It will be saved in a **.bmp**-file with the resolution 750 x 500 in the folder **picture\**. Here too you can choose between a public and a private folder.

### 5.1.4 *Cancel*



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

Shortcut: **Ctrl+Z**.

### 5.1.5 *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**.



### 5.1.6 *Font size*

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

### 5.1.7 *Instructions*



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

### 5.1.8 *Programming help*



Displays a text with a brief description of all the instructions necessary to accomplish the exercises of the current chapter. Shortcut: **F2**.

### 5.1.9 *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.

### 5.1.10 *Cancel*

Quits the editor without compiling but all modifications are saved.

### 5.1.11 Compile



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

### 5.1.12 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.

### 5.1.13 Pause/continue



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

### 5.1.14 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 C++, C# and Java™. It has been designed specially for CeeBot and is very well adapted to teaching purposes.

In CeeBot3, only a limited number of concepts are taught. First the students learn what a linear instruction sequence is, then learn to use loops that repeat a bloc of instructions a certain number of times. They will learn how to use functions without parameters, then functions with one single `float` parameter, and finally `for` loops. With these few concepts the students can program amazing animations.

However the compiler-interpreter used in CeeBot3 supports all the characteristics of a complete object-oriented language. In order to get a description of all the features of the language, open the instructions of any exercise in Chapter 25 **Project** (or of Chapter 6 in the demo version), then click at the end of the text on the link **CeeBot language reference**.

## 6 Using CeeBot3 in a classroom

The best way to get an idea about how CeeBot3 can be used in a classroom is to take a look at one exercise after another. When you select the user name **Teacher**, 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 the 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. CeeBot 3 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.

### 6.2 *CeeBot and creativity*

One fundamental aspect of CeeBot3 is to stimulate the creativity of the students. Even in exercises that give a precise task to the students, it is often better not to be too strict about the given instructions. As long as the student understands and is able to apply the concept that is the objective of the exercise, the goal of the exercise is reached, even if the drawing or the animation that he or she makes does not correspond exactly to what was required. It is of course useful to point out the difference between the instructions and the result to the student, because one general objective is to learn to stick to written instructions.

However the importance of creativity does not necessarily mean that all the students must concentrate their efforts on the free exercises. For example students who progress at a fast pace can very well leave aside those exercises that require much time, so that they can more quickly reach the more advanced exercises that will motivate them much more, or the project of chapter 25.

You should still be aware of the dangers of too fast a progression, which can result in a superficial comprehension of the different concepts. The student can get the impression that he or she fully understands, but he or she would be unable to apply the concept in a creative way. The student will then be blocked in his or her progression by a series of accumulated gaps in his or her knowledge, which must be dealt with by going back to the beginning, which is always frustrating.

# 7 Installing CeeBot in a classroom

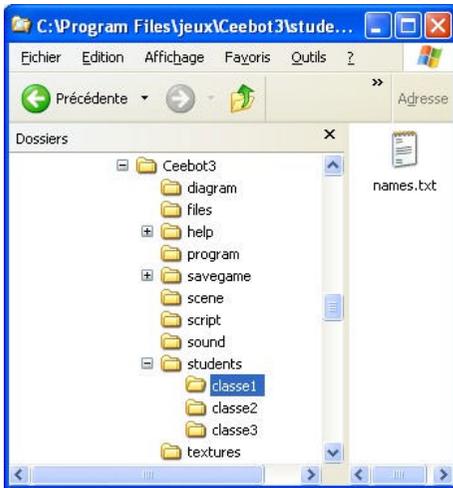
In order to install CeeBot in a computer classroom, you need to perform a few extra actions in order to enter the names of the students of the different classes and possibly to set up its use on a LAN.

## 7.1 Entering the students' names

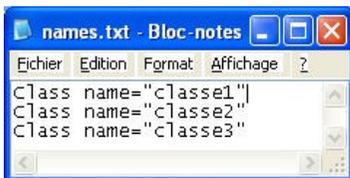
For now the way of entering the students' names is a little awkward, and it will be improved shortly.

In the folder where CeeBot3 is installed there is a folder called `\savegame\`. This folder contains all the programs that the students write (they are automatically saved), the pictures that result from their programs (saved with the instruction `save ( ) ;`), and all the files saved into a private folder.

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



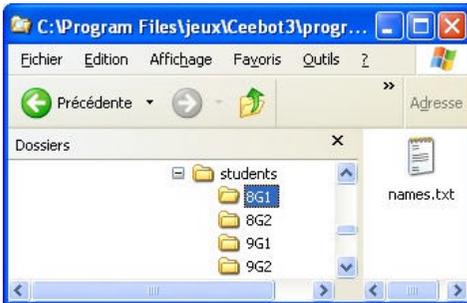
Each of these folders contains a file called **names.txt**. The folder `\students\` itself also contains such a file. If you open the file `\students\names.txt`, you will find there the list of all the classes :



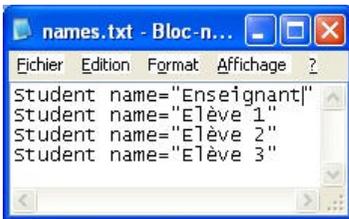
You can change the names of the classes in quotation marks, or add a class by simply adding a new line:



Then you will have to create the corresponding folders (menu **File - New - Folder**), and put into each new folder a file **names.txt**:



Now the classes are ready. You must now enter the names of the students for every class. Open the file **names.txt** in the folder of one of the classes:



Then change the names in quotation marks and add the necessary lines, so that the file contains the names of all the students of the class. If you add a line with **Student name="Teacher"**, you can access on this computer the solutions of the exercises and the settings. In order to avoid the students accessing the solutions, you can then delete this line if you wish.

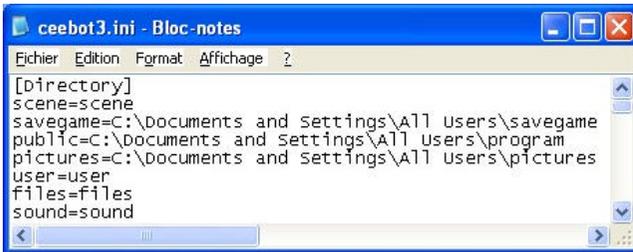
The next time that you launch CeeBot3, the classes and the students that you entered will appear in the start dialog. Every time a student writes a program or executes a program with the instruction `save ( ) ;`, the corresponding file will be saved in a similar folder structure in the folders **savegame\** (for the programs) and **savetempl** (for the pictures that are automatically saved with the instruction `save ( ) ;`).

## 7.2 Setting up the folders

The first time you launch CeeBot3, a file **ceeboot3.ini**, containing all the settings, will be created in the folder where you installed CeeBot3. The first lines in particular contain all the settings for the folders where the program will read and write its files:



You can enter here another folder path, on the same computer or on a server. Under Windows XP in user mode, the programs can not write into the folder **Program Files**. You must therefore relocate all the folders where files are written to another folder, for example the folder that is shared between all users:



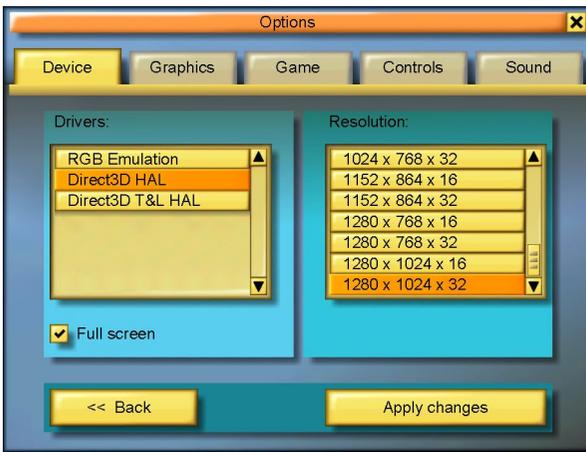
The folders defined with **scene=** and **user=** are not used in CeeBot3. The folder **savegame=** contains the programs made by the students, which are saved automatically. It also contains the pictures and programs saved into a private folder. The folder **savetemp=** contains the pictures saved with the instruction `save()`; . If you program an animation, several hundreds of pictures will be saved to this folder, therefore you should leave it on the local hard disk and not move it to the server. If this folder becomes too big, you can simply delete it. The folder **public=** contains the programs saved with the button **Save** into the public folder. The folder **pictures=** contains the pictures saved into the public folder. The folder **files=** is only used by the file management instructions, which are fully functional in CeeBot3, but are not mentioned in any of the exercises. The folder **sound=** is the place from which CeeBot3 will take the sounds that are integrated with the instruction `sound()`; into the animations. If you move this folder (for example so that the students can copy there sounds that they have recorded), you must also move the **.wav** files that are contained in the original **sound\** folder.

## 8 Options

Click **Options** to access various settings, separated into 5 categories. This button appears 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!

## 9 Development team

Daniel Roux, Denis Dumoulin, Otto Kölbl, Michael Walz

EPSITEC SA, Mouette 5, CH-1092 Belmont, Switzerland  
[info@ceebot.com](mailto:info@ceebot.com), [www.ceebot.com](http://www.ceebot.com), [www.epsitec.ch](http://www.epsitec.ch)