



1 Introduction

1.1 Configuration requise

CeeBot requiert un ordinateur moderne et puissant pour fonctionner. La carte graphique 3D a une énorme influence sur les performances. Cela pose parfois des problèmes avec les ordinateurs portables.

- o Processeur 300 MHz
- o 64 Mb RAM
- o Carte graphique 3D avec 16 Mb RAM
- o 100 Mb d'espace libre sur le disque dur
- o Windows[®] XP, 2000, ME, 98 ou 95

Les cartes graphiques suivantes sont recommandées :

- o nVidia GeForce 2, 3 ou 4
- o Matrox G400, G450 ou Parhelia

Les cartes 3dfx Voodoo ne sont pas officiellement supportées, bien que cela puisse fonctionner (sans garantie).

1.2 Installation

- o Insérez le CD-ROM CeeBot-Teen dans votre lecteur

Normalement, l'*autorun* s'active automatiquement après une dizaine de secondes. Cliquez simplement sur « **Installer** ». Si ce n'est pas le cas, procédez comme suit :

- o Double-cliquez sur **Poste de travail**.
- o Double-cliquez sur **(D:)** ou sur la lettre correspondant au lecteur de CD-ROM.
- o Double-cliquez sur **install.exe**.

Si ce n'est pas déjà fait, le programme vous demande d'installer DirectX[®] 8a.

Si vous avez déjà installé une version précédente de CeeBot, une nouvelle installation dans le même dossier détruira toutes vos sauvegardes.

1.3 Désinstallation

- o Double-cliquez sur **Poste de travail**.
- o Double-cliquez sur **Panneau de configuration**.
- o Double-cliquez sur **Ajout/Suppression de programmes**.
- o Double-cliquez sur **CeeBot-Teen** dans la liste.

2 Premiers pas

Dans ce chapitre, nous allons effectuer le deuxième exercice, pas à pas. Les explications détaillées viennent dans les chapitres suivants. Il s'agit juste ici d'un premier contact, pour vous familiariser avec CeeBot-Teen.



La première question posée par CeeBot est le nom du joueur. Il est important de donner son nom, car chaque utilisateur verra sa progression automatiquement sauvegardée. Donc, tapez simplement votre prénom ou votre pseudo la première fois. Par la suite, il suffira de choisir dans la liste. Lorsque c'est fait, cliquez « **D'accord** ».

Cliquez ensuite sur ce bouton :



CeeBot affiche maintenant la liste des exercices disponibles. La partie gauche correspond aux chapitres, et celle de droite aux exercices du chapitre sélectionné.

Lorsque vous aurez terminé le premier exercice, le deuxième apparaîtra dans la liste de droite.

Après avoir choisi « **1: Bouge** » et « **1: Juste avancer** », cliquez sur le bouton « **Jouer** ».



La lecture attentive de cet écran vous explique ce que vous devez faire. Lorsque vous avez compris, cliquez sur le bouton en bas à gauche pour sortir.



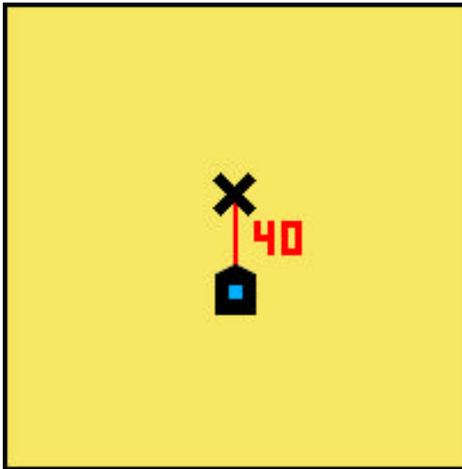
Si nécessaire, vous pouvez revenir dans cet écran en tout temps, en appuyant simplement sur « **F1** ».



Le robot miniature est posé sur le sol, dans une cave encombrée d'objets hétéroclites.

Les flèches du clavier permettent d'observer les alentours du robot.

Dans ce premier exercice, il faut programmer le robot pour qu'il avance tout droit de 40 centimètres.



La carte miniature en bas à gauche de l'écran résume la situation. Le symbole en bas représente le robot. La croix en haut est le but à atteindre. Les deux symboles sont distants de 40 centimètres.



Si ce n'est pas déjà fait, sélectionnez le programme numéro 1 en haut de la liste. Chaque robot peut recevoir 10 programmes, ce qui peut être utile pour faire des essais. Tous ces programmes sont automatiquement sauvegardés lorsque vous quittez l'exercice.

Le programme numéro 4 contient la **solution**. Essayez de ne pas le regarder pour l'instant !



Pour entrer dans l'éditeur de programme, cliquez l'icône {..}.



L'écran doit alors montrer un squelette de programme. Le curseur est directement positionné là où vous devez entrer les instructions du programme.

Ne touchez pas au texte `extern void object::Go()`, ni aux accolades.



L'exercice consiste à avancer le robot de 40 centimètres, jusqu'à la croix bleue. Pour cela, il suffit d'entrer une seule instruction :

```
move(40);
```

Tous les caractères sont en minuscules. N'oubliez pas le point-virgule qui termine l'instruction.



Cliquez alors sur le bouton « D'accord ». Si vous avez fait une faute, les caractères erronés sont mis en évidence en bleu, et un message d'erreur apparaît en bas.

Dans l'exemple ci-contre, c'est le point-virgule après la première instruction qui manque.

Lorsque tout est juste, l'éditeur se ferme.



Cliquez sur ce bouton pour exécuter le programme. Normalement, le robot doit avancer juste de la bonne longueur.

Et voilà, le premier exercice est réussi et terminé.

3 Menu principal



3.1 Programmation

Cette partie de CeeBot-Teen contient tous les exercices de programmation. Différents chapitres présentent des sujets de plus en plus compliqués. Vous devez effectuer les exercices dans l'ordre.

3.2 Jeu libre

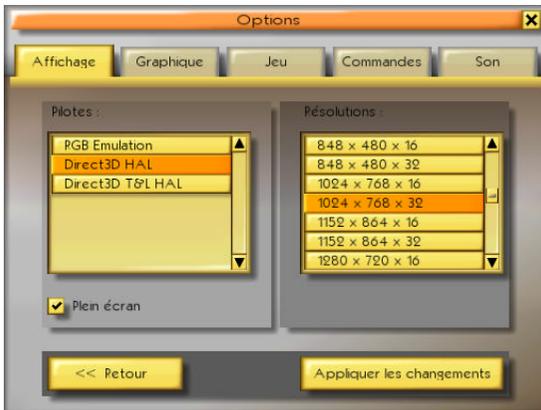
Le jeu libre donne accès à trois exercices spéciaux. « Déplacement libre », « Dessin libre » et « Programmation magique ». Ce dernier permet de créer des programmes par l'exemple. Le robot est déplacé librement pour dessiner, et le programme correspondant est créé au fur et à mesure.

3.3 Options

Le bouton « Options » accède aux réglages de CeeBot. Cinq onglets permettent d'atteindre les différentes catégories.

3.3.1 Affichage

Après l'installation, CeeBot utilise la résolution 640 x 480 x 16. Sur beaucoup d'ordinateurs, il est possible d'augmenter cette résolution, ce qui améliorera grandement la qualité graphique.



Pilotes :

Il faut choisir un pilote qui porte la mention HAL (Hardware Abstraction Layer). Evitez les pilotes ayant la mention « Emulation » ou « T&L ». Il arrive fréquemment que des pilotes avec des noms différents aient des comportements identiques.

Résolutions :

Les premier et deuxième chiffres indiquent le nombre de pixels en largeur et en hauteur dans l'écran. Le troisième chiffre indique le nombre de couleurs affichables (16 pour 65'000 couleurs et 32 pour 4 millions de couleurs).

Plus la résolution est grande et plus le jeu est beau. Mais il risque de devenir lent. Commencez par mettre 640 x 480 x 16. La plupart des cartes graphiques modernes supportent 1024 x 768 x 16. A vous d'essayer le meilleur compromis.

☞ Plein écran

Normalement, CeeBot occupe tout l'écran, quelle que soit la résolution. Si vous enlevez la coche, CeeBot tournera dans une fenêtre fixe ayant approximativement 640 x 480 pixels.

[Appliquer les changements]

Il faut cliquer ce bouton pour que les changements effectués dans cette page prennent effet.

3.3.2 Graphique

Si le jeu est saccadé, il faut essayer de supprimer certaines options graphiques jusqu'à ce que l'affichage soit fluide. A l'inverse, si tout est parfaitement fluide, essayez d'augmenter la profondeur de champ à 200%.



Quantité de particules (0% à 200%)

Les particules servent à simuler la poussière, la fumée, les éclats, les bulles sous l'eau, etc.

Profondeur de champ (50% à 200%)

La profondeur de champ détermine jusqu'où porte votre regard. Cette profondeur est de toute façon très différente selon l'atmosphère de la planète. Une grande valeur (par exemple 200%) permet de voir loin, mais nécessite une bonne carte graphique 3D.

Détails des objets (0% à 200%)

Lorsqu'un objet est au loin, il est représenté avec moins de détails. Une grande valeur éloigne la distance à laquelle le changement est effectué.

Nombre d'objets décoratifs (0% à 100%)

Ce nombre détermine la quantité d'objets décoratifs présents, tels que les plantes, les arbres, les cristaux, etc.

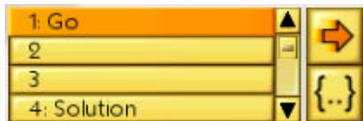
4 L'écran

Pendant un exercice, l'écran ressemble à ceci :



- 1 Les 10 programmes du robot
- 2 Les boutons pour bouger la caméra
- 3 Les boutons divers
- 4 La carte miniature
- 5 L'accès au menu

4.1 Les 10 programmes



Pour les robots, la partie gauche donne accès aux 10 programmes. Les 4 premières lignes donnent les noms des 4 premiers programmes. Le quatrième programme contient généralement la **solution** de l'exercice. L'ascenseur permet d'accéder aux 6 programmes suivants.



Ce bouton exécute ou stoppe le programme sélectionné.



Ce bouton édite le programme sélectionné (voir chapitre 6.1). Pendant ce temps, le jeu est en pause. Vous pouvez donc tranquillement réfléchir et modifier le programme.

4.2 La caméra

Les touches flèches du clavier ou les 4 boutons flèches permettent d'observer les alentours en bougeant la caméra. Les flèches « haut » et « bas » approchent ou reculent la caméra. Les flèches « gauche » et « droite » effectuent une rotation autour du robot.

4.3 Les boutons divers



Le **SatCom** est un petit appareil bien pratique qui donne toutes les informations sur l'exercice en cours. La touche « F1 » a le même effet.

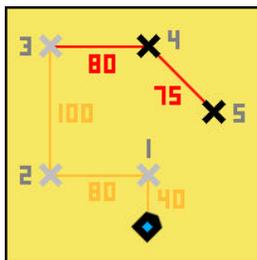


Ce bouton permet de changer le point de vue de la caméra. Certains robots ont 2 points de vue, d'autres 3. La touche « barre d'espace » a le même effet.



Ce bouton remet l'exercice dans l'état initial. Très utile lorsqu'un programme n'a pas eu l'effet souhaité, pour remettre les robots et les objets dans leurs positions de départ et effacer le dessin.

4.4 La carte miniature



La partie inférieure droite est une carte miniature, qui résume la situation avec une vue aérienne. Les croix représentent des points de passage. Les distances sont toujours exprimées en centimètres.

Le « rectangle pointu » représente le robot, la pointe étant tournée vers l'avant.

4.5 Le menu



La petite croix tout en haut à droite affiche le menu suivant :

- | | |
|--------------------|--|
| Continuer | Ferme le menu et continue l'exercice. |
| Options | Accès aux réglages du jeu. Certaines options ne sont pas accessibles. Pour avoir accès à toutes les options, il faut quitter l'exercice en cours, et choisir « Options » depuis le menu principal (voir chapitre 3.3). |
| Recommencer | Annule l'exercice en cours pour le recommencer depuis le début. |
| Abandonner | Annule l'exercice en cours et retourne à l'écran des exercices. |

5 Le SatCom

Le **SatCom** est un petit appareil bien pratique, qui affiche toutes les instructions sur l'exercice en cours, et qui contient aussi la base de données sur le langage de programmation CeeBot.



Il fonctionne un peu comme un navigateur Internet. Chaque fois qu'un mot est souligné en bleu, vous pouvez le cliquer pour afficher la page qui traite de ce sujet.



Affiche la page précédente.



Affiche la page suivante.



Affiche la page initiale, qui était visible lorsque le **SatCom** a été ouvert.



Copie tous les caractères sélectionnés dans le bloc-note. Il est ainsi possible de copier un programme donné en exemple, pour ensuite le coller dans l'éditeur de programme (voir paragraphe 6.1.4).



Affiche les instructions sur l'exercice en cours. La touche « F1 » ouvre toujours directement cette page.



Affiche un petit dictionnaire « anglais-français » des termes rencontrés dans CeeBot. Lorsqu'un terme anglais est écrit avec une majuscule, c'est qu'il doit s'écrire ainsi dans un programme.



Affiche l'aide générale sur le langage de programmation CeeBot. C'est une véritable mine d'or, qui explique en détail la syntaxe de chaque instruction. La touche « F2 » ouvre toujours directement cette page.



Eteint le **SatCom**. Sa fenêtre est donc fermée.

Lorsque l'un de ces boutons est gris, cela signifie qu'il n'a aucun effet dans la situation actuelle.

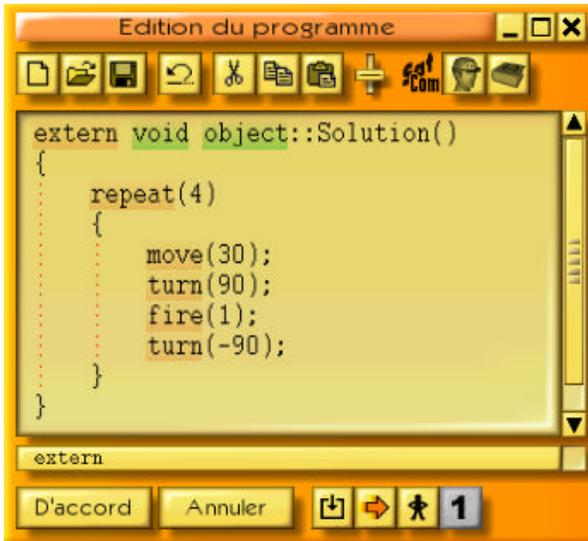
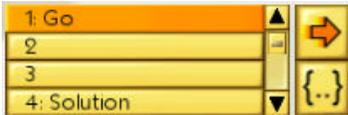
6 Programmation

Tous les robots sont programmables.

6.1 L'éditeur de programmes

Pour entrer dans l'éditeur de programme, il faut :

- ?? Choisir le programme à éditer parmi les 10 à disposition.
- ?? Cliquer le bouton `{..}` « édite le programme sélectionné ».



La fenêtre de l'éditeur peut être déplacée en tirant la barre de titre avec la souris. Elle peut également être redimensionnée en tirant les bords ou les coins. La prochaine fois que vous ferez appel à l'éditeur, la fenêtre sera à la même position et aura les mêmes dimensions.

-  **Taille réduite**, pour réduire la fenêtre à une barre tout en bas de l'écran.
-  **Taille maximale**, pour agrandir la fenêtre sur tout l'écran.
-  **Fermer**, équivalent au bouton « D'accord ».

Si vous entrez dans l'éditeur pendant que le programme s'exécute, le jeu n'est pas mis en pause. Cela permet alors d'observer le déroulement du programme (voir chapitre 6.1.11).

Les mots clés du langage apparaissent en couleur, pour faciliter la compréhension :

Couleur	Nature	Exemple
Orange	Instruction	move, turn, pendown, repeat, etc.
Vert	Type d'une variable	int, object, etc.
Rouge	Constante, catégorie	Red, Scrap, Metal, etc.

Lorsque le curseur est sur un mot clé du langage, la barre sous la fenêtre d'édition résume succinctement la syntaxe. Un clic sur cette barre appelle le **SatCom** pour vous donner plus de précisions. La touche « F3 » a le même effet.

Un double-clic sur un mot le sélectionne rapidement. Si le bouton de la souris est maintenu pressé, les caractères sont mis en évidence. « Shift+flèche » met également en évidence quelques caractères. Les touches flèches actionnées avec la touche « Ctrl » permettent de se déplacer par mots. « Ctrl+Shift+flèche » met en évidence par mots entiers.

6.1.1 Nouveau



Efface tout le programme en cours d'édition, et crée un squelette vide de la forme :

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

Nouveau est le nom du programme, que vous pouvez remplacer par un nom de votre invention, mais sans espaces. Par exemple *Cherche*, *JoliDessin*, *PetiteFleur*, etc.

Remarque : Si vous avez cliqué ce bouton par erreur, il suffit d'annuler l'action.

6.1.2 Ouvrir et Enregistrer



Tous les programmes que vous créez sont automatiquement sauvegardés dans l'exercice en cours. En revanche, si vous désirez réutiliser un programme dans un autre exercice, il faut l'enregistrer explicitement, pour pouvoir le rouvrir par la suite.

Privé Le programme est mis dans un dossier privé, qui dépend donc du nom donné au début du jeu.

Public Le programme est mis dans un dossier commun à tous les joueurs.

Juste en dessous de la barre de titre du dialogue apparaît le nom du dossier. Par exemple, **SavegameJoueur\Program** signifie que le dossier est généralement **c:\Program Files\CeeBot-Teen\SavegameJoueur\Program** (selon le dossier dans lequel vous avez installé le jeu). Cela vous permet de récupérer un programme dans une application extérieure à CeeBot. Par exemple pour envoyer un programme par e-mail à un ami.

Les touches « Ctrl+O » et « Ctrl+S » permettent d'accéder rapidement aux fonctions *Ouvrir et Enregistrer*.

6.1.3 Annuler



Annule la dernière action effectuée dans le programme. Il est ainsi possible d'annuler en tout temps les 20 dernières modifications.

Le touche « Ctrl+Z » a le même effet.

6.1.4 Couper, Copier et Coller



Coupe ou *copie* les caractères mis en évidence, pour pouvoir les *coller* ailleurs. Si aucun caractère n'est mis en évidence, c'est toute la ligne qui est prise en compte.

Les touches « Ctrl+X », « Ctrl+C » et « Ctrl+V » permettent d'accéder rapidement aux fonctions *Couper*, *Copier* et *Coller*.

6.1.5 Taille des caractères

Ce petit curseur permet de choisir la taille des caractères pendant l'édition et dans le **SatCom**. Cette taille est mémorisée jusqu'à ce qu'elle soit changée à nouveau.

6.1.6 SatCom : Instructions



Donne les objectifs de l'exercice. Vous pouvez aussi appuyer sur « F1 ».

6.1.7 SatCom : Aide à la programmation



Donne les instructions générales sur la programmation, à partir desquelles vous avez accès à toutes les explications spécifiques par des liens hypertextes. Vous pouvez aussi appuyer sur « F2 ».

6.1.8 D'accord

Compile le programme puis quitte l'éditeur si aucune erreur n'est détectée. En cas d'erreur, la partie incriminée est mise en évidence et la barre sous la fenêtre d'édition indique la nature de l'erreur.

6.1.9 Annuler

Quitte l'éditeur sans compiler le programme. Toutes les modifications effectuées dans le texte sont néanmoins enregistrées.

6.1.10 Compiler



Compile le programme, sans quitter l'éditeur. Cela vous permet de vérifier si le programme contient ou non des erreurs.

6.1.11 Démarrer/stopper



Démarré ou stoppé le programme, sans quitter l'éditeur. Cela permet de faire du *debug*, c'est-à-dire de chercher les causes d'un éventuel mauvais fonctionnement. Lorsque le bouton de droite représente un homme avec les mains sur les hanches, le programme est exécuté en *pas à pas*, c'est-à-dire instruction par instruction.

Remarque : Lorsque le programme est en exécution, la fenêtre d'édition devient orange. Vous pouvez alors observer le déroulement du programme, mais vous ne pouvez plus le modifier.

6.1.12 Pause/continuer



Passé alternativement du mode *pas à pas* au mode sans interruption.

6.1.13 Un pas



Exécute l'instruction suivante, en mode *pas à pas*. La partie sous la fenêtre d'édition montre alors le contenu des différentes variables du programme. Ce contenu change à chaque pas, en fonction des affectations effectuées.

6.2 Le langage CeeBot

Le langage CeeBot est très proche de Java[?], de C++ et de C#. Il est adapté à une approche pédagogique. Dans ce manuel, nous ne présentons que quelques exemples simples. Pour avoir une description complète, utilisez le **SatCom** en appuyant sur « F2 ».

Attention de respecter les majuscules et les minuscules. Par exemple, si vous écrivez `Move`, le programme ne sera pas compilé. Il faut écrire `move`.

Chaque instruction est terminée par un point-virgule ;.

Un commentaire commence par `//` et se termine à la fin de la ligne. Il n'a aucune incidence sur le déroulement du programme, mais sert uniquement à le rendre plus explicite.

6.2.1 L'instruction `move`

Utilisez l'instruction `move()` ; pour avancer ou reculer d'une distance exprimée en centimètres. Un nombre négatif permet de reculer. Le robot n'essaie pas d'éviter un obstacle avec cette instruction.

```
move(40); // avance de 40 centimètres
move(-10); // recule de 10 centimètres (négatif)
```

6.2.2 L'instruction *turn*

Utilisez l'instruction `turn()` ; pour faire tourner le robot d'un certain nombre de degrés sur lui-même. 90 degrés signifie un quart de tour à gauche, 180 degrés signifie un demi-tour, 360 degrés signifie un tour complet. Voici quelques exemples avec `turn()` ; :

```
turn(90); // quart de tour à gauche
turn(-90); // quart de tour à droite (négatif)
turn(180); // demi-tour
```

6.2.3 L'instruction *pendown*

Cette instruction abaisse l'un des huit crayons du robot dessinateur. A partir de cet instant, tous les déplacements avec `move` laisseront une trace. Une rotation avec `turn` ne dessine rien, car le crayon est exactement au centre du robot.

```
pendown(Red); // abaisse le crayon rouge
pendown(Blue); // abaisse le crayon bleu
```

Les huit crayons disponibles sont :

Black	Crayon noir
Yellow	Crayon jaune
Orange	Crayon orange
Red	Crayon rouge
Purple	Crayon violet
Blue	Crayon bleu
Green	Crayon vert
Brown	Crayon brun

6.2.4 L'instruction *penup*

Cette instruction relève le crayon du robot dessinateur. Il ne faut pas donner de couleur entre les parenthèses, mais il ne faut pas oublier de les écrire tout de même.

```
penup(); // relève le crayon
```

6.2.5 L'instruction *repeat*

Cette instruction permet de répéter un groupe d'instructions situé dans les accolades { } qui suivent. Attention, l'instruction `repeat` n'est pas terminée par un point-virgule !

```
repeat (4)
{
    move(20);
    turn(90);
}
```

6.2.6 L'instruction *fire*

L'instruction `fire()` ; permet de faire feu avec le canon du robot. En général, on tire des rafales d'une seconde :

```
fire(1);
```

6.2.7 Les instructions *grab* et *drop*

L'instruction `grab()` ; permet de saisir avec la pince du bras manipulateur un objet qui se trouve au sol. L'instruction `drop()` ; dépose l'objet actuellement transporté. Voici par exemple comment saisir ce qui se trouve devant le robot et le poser 15 centimètres plus loin :

```
grab(); // prend l'objet
move(15); // avance de 15 centimètres
drop(); // dépose l'objet
```

6.2.8 L'instruction *find*

Avec l'instruction `find()` ;, le robot peut chercher des objets comme des déchets ou des usines. Donnez entre parenthèses le nom de l'objet que vous cherchez. Si l'objet existe, le robot s'y rend immédiatement, en évitant les obstacles éventuels. Voici par exemple comment chercher le déchet le plus proche :

```
find(Scrap);
```

6.2.9 Les instructions *if* et *else*

L'instruction `if() {}` permet d'exécuter des instructions seulement à une certaine condition. La condition est donnée entre parenthèses `()`, les instructions entre accolades `{}`.

Attention, l'instruction `if` n'est pas suivie d'un point-virgule. L'instruction `else` n'a ni point-virgule ni parenthèses.

Voici par exemple comment aller vers un désintégrateur si l'objet saisi est en métal, et vers un destructeur dans le cas contraire :

```
if( load.material == Metal )
{
    find(Disintegrator);
    drop();
    move(-10);
}
else
{
    find(Destroyer);
    drop();
    move(-20);
}
```

6.2.10 L'instruction *detect*

Avec l'instruction `detect`, le robot peut voir devant lui s'il existe un objet précis.

```
if ( detect(Barrier) )
{
    turn(90);
}
else
{
    move(20);
}
```

7 Suite

CeeBot-Teen est une initiation à la programmation. De nombreuses autres instructions permettent d'apprendre des notions de programmation plus complexes. Essayez CeeBot-A, qui est la suite logique de CeeBot-Teen.

8 Développeur

Daniel Roux, Denis Dumoulin, Otto Kölbl, Michael Walz

EPSITEC SA, Mouette 5, CH-1092 Belmont
ceebot@epsitec.ch, www.epsitec.ch