RoboMind Course plan Primary 1

Courseplan

Overview of the RoboMind lessons

General info

Subject	**RoboMind Course plan Primary 1**
Level	For children aged 9 years or older
Prerequisites	Basic computer skills (file opening and saving, simple text editing)
Materials	Presentations, exercises and hints (pdf), movies (mp4), software, scripts and maps (RoboMind)
Duration	7 weeks, 1.0-1.5 hours per week

Goal

The aim is to gain insight into logic, automation and robotics by programming a virtual robot with the programming language ROBO. This will also give direct insight into the operation of technical appliances as they are all around us.

ROBO is a small language with a concise set of rules where no prior knowledge is required, so students can get started straight away.

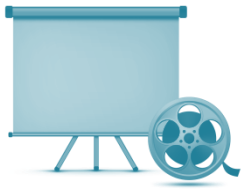
As the children make the exercises, they get acquainted with the possibilities and impossibilities of the programming language and acquire more insight into the power of logic. They will also gradually learn how a large problem can be solved by breaking it up into smaller pieces which can be solved more easily.

This course specifically teaches children the programming concepts regarding repetition and conditions. Conditions ("if ... then ..." rules) prove to be quite challenging and abstract for children in this age group. To ensure a minimum level of understanding, we have included an interactive quiz in the presentations around lesson 5 and 6.

During the last lesson the students do a kind of 'masterpiece'. This involves an exercise around a problem which seems very difficult to solve at first. However, the problem is actually rather easy to solve if it is subdivided into smaller sub problems. The children should work out for themselves how this problem may me put into a set of smaller sub problems which they can do by making good use of what is learned in the previous lessons.

General structure of the lessons

All lessons have the following structure.



Every lesson starts with a short multimedia presentation which highlights some aspects of what robots do en what their role is in society. In addition a brief explanation is given about the programming instructions used in the exercises and why these could be useful. This part takes about 15-20 minutes.

Next, the children will work for themselves with the virtual robot 'Robo' in RoboMind. They will be asked to let Robo do tasks as set out in the exercises. There should be no more than two children behind each computer.

It is important to let the children solve the problems mostly by themselves. Robo is a virtual robot, so you can try as often as you want, and you can never do wrong. However, the teacher can always offer new clues (provided in a separate document) if a problem proves too difficult to solve. Thus basically every problem can be solved eventually by everyone. This part takes ± 45 minutes.

The more exercises a student finishes, the better. However, it does not matter for the series of lessons if not all exercises have been completed. For quick learners, we provide additional exercises toward the end of the programming activity, such that the smart ones do not get bored.

After the programming activity a brief discussion of 5 to 10 minutes is recommended. Here the children can be given the opportunity to tell what was fun and what was considered to be (very) challenging. This can then be given some extra attention next time.

Planning over the weeks

Week 1: General introduction. What types of robots are there? How can they work for us? We creatively explore RoboMind by programming a robot dance. At first we will use the remote control in the program. Then we will write the commands for Robo ourselves (the real programming) as we will continue to do during the rest of this course. It is important for the children to see that using the remote control seems less work, but that programming yourself is much handier in the long run.

Week 2: Guarding and slalom. Question is how much work Robo can do for us, without having to write a large program. We let the robot do a guarding task which requires Robo to drive around a lot. We show how this can be concisely programmed with just a few lines. Furthermore, we show that a good program can be applicable at multiple sites. The children should furthermore develop insight into where the exercises differ and where they are equal. Where equal, the same (piece of) programming code can be re-used. This saves a lot of programming!

Week 3: Painting floors. Regular patterns can be easily automated. We introduce the "`repeat`" instruction which will be used to let Robo paint all kinds of floors. We demonstrate the power of smart programming by showing that the same program can be used to paint a small floor as well as a large ballroom floor.

Week 4: Factories. Repetitive tasks can sometimes be performed better by a machine. This is the case in factories where much of the simpler tasks are automated. Again, the "`repeat`" instruction proves useful. The exercises are intended to further practice with `repeat`, but also to show that what Robo does is not really very different from what happens in a factory. Furthermore, students should gain insight into which (parts of) problems are suitable for programming with the "repeat" instruction. The last assignment is a challenging one for the fastest learners and aims to deepen this understanding even somewhat more.

Week 5: Navigating. How do we find our way in unfamiliar territory? Thanks to sensors onboard the robot, we can determine in what condition we are. We introduce programming with conditions ("if ... then ..." constructions) and do a quiz during the presentation to make sure that everyone understands this well enough. In the assignments, the students will discover that Robo can solve difficult tasks by repetitively taking only small steps based on simple choices (the conditions). This gives a first insight into the "divide and conquer" strategy in which you solve a problem by dividing it into smaller pieces that can be solved more easily. For the fastest, there is a last assignment which requires an extra condition within the repeat loop (from 2 to 3 conditions).

Week 6: Tracking and Tracing. How do we find our way in unfamiliar territory? As during the previous lesson, we use the sensors onboard the robot to do path finding. We practice once more with the interpretation of conditions as we learned them during the previous lesson. During the presentation there is a quiz to make sure that everyone understands how Robo will behave when conditions are true or false. In the exercises the number of conditions in the loop is expanded and the children are asked to program Robo such that he will find its way by tracking dots in the various maps. The aim is to show that with simple conditions complicated behavior can emerge which is even robust against changes in the environment (the maps will be used).

Week 7: Line following. We have seen how and where we can use the repeat and conditions constructs to do tasks and respond to the environment. With this knowledge, we can program a

robot that follows lines: a task which has to be performed in many robot competitions! Solving this last week's assignment is a sort of 'masterpiece' with which the children demonstrate what they have learned during the previous weeks. After the programming part, there is a small recapitulation with the children about what they think that has been learned during the course. At the end everyone receives a RoboMind Programming Certificate which concludes the course.

Performance monitoring

The lessons have an open training objective and are oriented towards problem solving. Main objective is to gain an understanding of logic, automation, robotics and how you can solve bigger problems by breaking these down into smaller sub problems. This insight is necessary to solve the problems in the exercises. The number of hints or clues a student needs to solve a problem gives some insight into how much understanding is acquired. At least as important, however, is that students enjoy the lessons, come to see that technology is really fun, and that the resolution of a problem can give a lot of satisfaction.

Tips

Let the children do as much as possible by themselves, but give clues if needed, such that they have a least seen a bit of all exercises.