



**ROYAL INSTITUTE  
OF TECHNOLOGY**



**Stockholms  
universitet**

# A school student laboratory using robotics

---

Based on Lego Mindstorms

**Jenny Gustavsson**

Master thesis in technology and learning,  
degree project for the study programme  
Master of Science in Engineering and of Education

**Stockholm, 2012**

**Examiner**

Magnus Hultén, Stockholm House of Science

**Head supervisor**

Cecilia Kozma, Stockholm House of Science

**Assistant Supervisor**

Carolina Svensson-Huldt, Department of Mathematics and Science Education, SU

**External Supervisor**

Sebastian Reuter, IfU



## **Abstract**

This report presents a thesis done at the programme Master of Science in Engineering and Education at KTH and SU. This project aims to create a laboratory work trying to inspire students for the technical subjects (mathematics, physics, technology and computer science) and is conducted at the department IfU at RWTH University in Aachen, Germany. It is a part of the project RoboScope, developed in Germany to inspire young people to study more engineering. This laboratory work uses Lego Mindstorms and the students should work with robotics. The laboratory work is based on a theme; the students will build a robot that can complete a specific task, a rescue task in a nuclear power plant. The new parts in the laboratory work "Rescue Work" involve more interactivity, higher degree of freedom, older students and the programming is made in Java. The laboratory work uses the model of Brall. Evaluation of the laboratory work has been done together with colleagues at IfU and a group of school students.

## **Keywords**

Laboratory work, Degrees of freedom, Interactive process, STS, Lego Mindstorms, Robotics

## **Sammanfattning**

Denna rapport redovisar ett examensarbete gjort på programmet Civilingenjör och Lärare på KTH och SU. Examensarbetets syfte är att skapa en laboration för att försöka inspirera studenter till de tekniska ämnena (matematik, fysik, teknik och datavetenskap) och är genomfört vid institutionen IfU på universitetet RWTH i Aachen, Tyskland. Den är en del av ett projekt RoboScope, drivet i Tyskland för att inspirera fler elever till att studera till ingenjörer. Denna laboration använder sig av Lego Mindstorms och studenterna arbetar med robotik. Laborationen utgår från ett tema och studenterna ska bygga en robot som kan genomföra den specifika uppgiften de blir tilldelade. Uppgiften är en räddningsaktion i ett kärnkraftverk. De nya delarna i denna laboration involverar mer interaktivitet, högre frihetsgrader, äldre elever och programmeringen genomförs i Java. Laboration använder sig av modellen av Brall. En utvärdering är gjord tillsammans med kollegor på IfU och en grupp av studenter.

## **Nyckelord**

Laboration, Frihetsgrader, Interaktiv undervisning, STS, Lego Mindstorms, Robotik

## Wordlist

**Degree of freedom:** Schwab introduced the term “degree of freedom” 1962. It measures in three different aspects, how free a laboratory work is. These aspects are question, method and results, each of these can either be open or given.

**STS:** “Science, Technology, Society” or also called STS is one of Roberts (1994) emphasis. This emphasis means that we learn science to be able to make our own decisions.

**Autonomous robot:** means that the robot can control itself.

**Semi-autonomous robot:** means that the robot can in some parts control itself.

**Lego Mindstorms NXT:** Lego Mindstorms NXT is a programmable construction line from Lego. It consists of a NXT Intelligent Brick, sensors, motors and Lego parts.

**ZPD:** Zone of proximal development. A concept developed by Vygotskij and means the difference between what a student can learn without help and what the student can learn with help from other students or the teacher.

**Interactive whiteboard system:** A whiteboard that also can be used as projector and it is connected with a computer, so you can save everything that you have written on the whiteboard in the computer.

## **Preface**

The summer 2010 I lived and studied in Aachen, Germany. During those 5 months I had a lot of fun and learned a lot of German. But one semester was fast gone and I had the feeling that it was not enough. So I started to think about doing my thesis in Germany. Aachen was the obvious choice for me because I knew some people there and I knew the city. But it was not easy to find a Master Thesis project for Master of Science in Engineering and Education. But then I got a tip from a friend that I should try on the institutes ZLW/IMA & IfU; and there I found this Master Thesis project. I found it really interesting and it fitted perfectly with my Master programme.

The work with this Thesis has been a big experience for me. It has been a great experience for me to work in German and at the same time write in English. To see how it all works in Germany has been a great experience, especially since I have thoughts about working in Germany in the future. It was a new experience for me to work with robotics and Java. I have really enjoyed the work. The Thesis was started in May 2011 and was finished in January 2012.

I had a lot of help from my German Supervisor Sebastian Reuter and both the Swedish Supervisors Cecilia Kozma and Carolina Svensson Huldt, I really want to thank them all for their support. I also want to thank my classmates Agnieszka Woronin and Marcus Blomqvist for supporting me in my work with this thesis and all the time believing in me. A huge thanks to Sanna Jonsson for the support.

# Contents

1 Introduction .....	7
1.1 Aim.....	8
2 Background .....	9
2.1 Brall.....	9
2.2 Current Theme .....	10
2.3 Degrees of freedom .....	10
2.4 Interactive learning process .....	11
2.5 Interest.....	12
3 Robotics.....	13
3.1 History of Robotics .....	13
3.1.1 The three laws of Robotics.....	14
3.2 Robotics today .....	14
3.3 Robots in Education .....	15
3.3.1 Lego.....	15
4. The Results .....	17
4.1 Description of the “Rescue Work”.....	17
4.2 Experience from using “Rescue work” .....	18
5 Discussion .....	22
References .....	25
Figures.....	28
Appendix .....	29
A. Laboratory Material.....	29
A.1 Instruktionen für Assistenten /     Handledning till assistenterna .....	30
A.2 Presentations for the laboratory work .....	55
A.3 Building instructions.....	69
A.4 Handout Programmierung / Stencil programmering .....	91
A.5 Fragebogen / Utvärdering .....	103
B Rückgabe von Fragebogen / Resultat av utvärderingen .....	107

## 1 Introduction

This is the Master Thesis in the course SA210X at Master of Science in Engineering and of Education programme at KTH, Stockholm. This Thesis was written at an Institute from RWTH Aachen, the IfU (Institute für Unternehmenskybernetik). This Institute works alongside two other Institutes, ZLW and IMA. They work with many different projects and tasks with a focus on learning - and science management; software - and product engineering; cybernetics management. I have worked with one of their projects, RoboScope (fig 1).



Figure 1 RoboScope, logo for the RoboScope project.

In a couple of years there will be a great need for engineers in Germany (Bundesagentur für Arbeit, 2010). Already today the demand for educated engineers is greater than the number of engineering students at all German Universities. Take all examined Engineers between 25 and 34 Years old divided by all Engineers between 55 and 64 Years old. This ratio would be 0.9 for Germany and for Sweden the ratio would be 4.7 (Bundesagentur für Arbeit, 2010). The Germans have tried to make the younger students more interested in technical studies, and therefore there are many of different projects in Germany aiming to make the Institute students take part in those subjects. Many of the projects are specially designed for girls, since just 20% of girls in Germany are studying computer science (Fraunhofer, Roberta Reihe Band 1). For example the Fraunhofer IAIS (Institute for Analyze- and Information system) has developed a project to make girls more interested in Robotics, the Roberta project. The project of this Master Thesis within RoboScope is similar to the Roberta project but involves boys as well, and it is made to inspire the German students for studies in technical subjects.

Technical along with scientific subjects are in Germany called the MINT-subjects; MINT-Mathematics, Informatics, Science and Technical.

RoboScope is a part of a national project and was developed as an extracurricular learning venue with the aim of encouraging the students to be more interested in the MINT-subjects. RoboScope is a project for pupils (in which the students spend four hours building and programming a robot within the confines of a specific theme). After primary level, the German school system splits into several different school types, namely the Gymnasium, Realschule or Hauptschule. This project is directed at students in the gymnasium, which mean that they attend school until years 12 or 13.

The concept behind RoboScope is a laboratory project for the German upper school students where they work with Lego Mindstorms. With Lego Mindstorms there are many possibilities to build and program their own robot with specific tasks. The aim of RoboScope is to get the students more interested in the technical and scientific subjects but also topics such as medicine, social and ethical issues. At the same time the students get the possibility to practice their skills in critical thinking, problem solving, creativity, innovation, communication and cooperation. They also get to experience the university for the first time. Schools or other youth groups in NRW (North Rhine-Westphalia) within the RoboScope project have the possibility to register for a visit.



This project (RoboScope) is promoted by the Department for Innovation, Science, Research and Technology in NRW; as a part of the community initiative “Zukunft durch Innovation. NRW” (ZDI) (Future through Innovation). The aim of “Zukunft durch Innovation. NRW” (Future through Innovation) is to inspire the students to study science or engineering. In this way the Germans hope to avoid the lack of people educated in science and engineering in the future.

Since last summer (Summer 2010) the Institute has offered laboratory projects for students up to secondary school level. Behind the works they have what they call a didactic concept (the concept for how they are going to educate, and what I will call the “concept for education” in this work), and that concept has also been the base for this project.

In this thesis I develop a laboratory work called “Rescue work” for older students, in upper secondary school using Lego Mindstorms. The laboratory work “Rescue work” is aimed at students who have worked with programming in Java earlier. During the laboratory work the students get the possibility to become familiarized and work with the technical subjects and get new knowledge and experience. The students get the opportunity to experience life as an engineer.

Working with a laboratory work might have several positive effects on the students (creates interest in a subject, as well as nurturing skills in both work and study). During the laboratory work the students are given the possibility to develop solutions on their own.

## **1.1 Aim**

The aim of this project is to develop and evaluate a laboratory work at RoboScope. The development of the laboratory work is divided into two parts, one technical and one pedagogical. The purpose of the project is to make the German students in upper secondary school more interested in scientific subjects. It should inspire for future studies in the MINT-subjects.

## 2 Background

In this chapter some theoretical aspects are explained. Those aspects are used during the work with the laboratory work.

### 2.1 Brall

Dr. Stefan Brall has created a model which is based on the idea that experiences serve as an effective starting point for all kinds of learning processes (Brall, 2008). Brall posts that the most important thing is a permanent evolution of knowledge and skills. The model made by Brall consists of an experience-based learning cycle; it involves experience, reflection and development of new acting strategies. Dysthe (2003) implies that it is important for the students to be a part of a social network because such interaction and cooperation is necessary for the students learning process. Dysthe also believes that the students learn from the other students' experiences and thus construct new knowledge together. Yager (2005) holds the belief in order to increase understanding we have to make education more relevant to the students' lives.

The model created by Brall (fig 2) consists of learning based on the reflection of experiences and is connected to learning that generates actions. In the inner cycle are the students experiences (E) transformed into an experience-based model of their reality (EM) via a reflection (R). That means that the students make a model of their reality with help from reflection and experience. The outer cycle generates action (H) and with help from reflection creates it models for actions (GM).

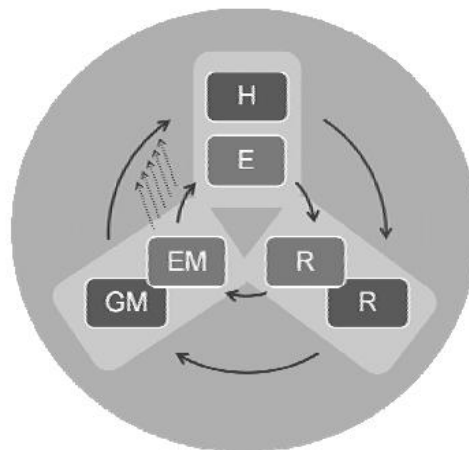


Figure 2 Description of the model from Brall, showing the connection between experience, reflection and action.

This theory model is already used in the existing RoboScope works and should also be used in the laboratory work "Rescue work". The main difference between the old ones and the new one is that the "Rescue work" involves more discussions to support the reflection and experience element. In the discussions the students work together to connect their works with their everyday lives and are able to reflect on their new experiences. The laboratory work itself allows for new experiences in a lot of different subjects, for example robotics, design, construction. Through building and programming their own robot the students also benefit from the practical side of things.

## 2.2 Current Theme

“Rescue work” is a socio-scientific approach and was chosen as a theme for the project, a current theme in our society today. The theme was chosen so that it should be something that the students can relate to and it shows the benefits of using robotics in such situations. This theme relates to the curriculum emphasis “Science, Technology, Society” or also called STS (Roberts 1994).

Roberts (2007) also describes two visions about the scientific education. In Vision one the knowledge of scientific concepts and the ability to solve scientific problems are central, it is a purely scientific activity (Roberts, 2007). Vision two works with a general capability to devise and solve problems, it can develop the ability to receive and solve problems but also make the people act different in the society. In the laboratory work “Rescue work” vision two is used. We can use robots in situations that are critical or dangerous for the humans. When working with robotics we have to consider other facts and circumstances as well and it will not just be pure scientific subjects. When the students work with building robots (action) they also get to see other sides of the scientific subjects such as design and engineering.

## 2.3 Degrees of freedom

RoboScope is a project focusing on scientific education within the realms of the laboratory works. Schwab J.J (1962) was the first one to discuss “The Degrees of freedom”. It measures in three different aspects, how free a laboratory work is. These aspects are hypothesis, method and results, each of which can either be open or given. Using these, Schwab defined the concept of degrees of freedom ranking from zero to three, where zero is when everything (hypothesis, method, result) is given and three is completely open.

Degree of freedom	Explanation
0	Hypothesis, method and result is given
1	Two of following are given: hypothesis, method and result
2	One of following is given: hypothesis, method and result
3	Hypothesis, method and result is open for the students to decided themselves

According to Löfdahl (1987) a high degree of freedom increases the students’ responsibility and freedom when they work with a laboratory work. Berg (2003) writes that students which have performed open laboratory work in higher degree of freedom have reflected during the work, which he means contribute to a deeper understanding. The students can more easily describe what they have done during the laboratory work. The students are asking reflective questions which indicates that they understand the theory. It also makes the students more positive and engaged in the laboratory work.

In the work with the “Rescue work” the plan was to use a high degree of freedom. The idea was that the students should make their own decisions because that can inspire the students more and they get more experience from making their own decisions. To get an exercise (or a hypothesis) can also make the students feel that they need to have the knowledge to complete the exercise (hypothesis). It can inspire them to search new knowledge. The former concept

for laboratory works at RoboScope was setting some limits for the degrees of freedom, then the students get a description how they should build “their” robot, what they should do and in some cases how they should do it. It was hard to get the laboratory work with a degree of freedom of three, but with some work it was possible to make a laboratory work with higher degree of freedom than zero. Referring to the degrees of freedom we can say that the question is given. The hypothesis (or problem) is to build and program a robot to do a specific exercise; but with the method we have the possibility to make a more open model. The students can in some ways choose their own way to make their robot. The result is in some parts open but in some parts already given. The students know what to build (a robot) and the exercise the robot should manage are also given, but how to program and what robot they choose to build are for the students themselves to decide.

Using a high degree of freedom gives possibilities for the students to use their own experience, reflection and action during the laboratory work “Rescue work”. The students can decide how to use their knowledge and then they can relate their work to their life and earlier experience. The students have the possibility to make their own decisions (and in that way also use their earlier experience and reflections) with high degree of freedom.

## 2.4 Interactive learning process

The image of education and learning has changed during the years. Today we say that knowledge is actively constructed (Barr and Tagg, 1995). Students achieve a more profound education when they successfully construct knowledge and then retain this constructed knowledge. When students actively develop their skills, they are constructing knowledge and achieving a deeper understanding. When the students work in groups they use their language to defend their ideas which would promote construct understanding (Syh-Jong, 2007).



Figure 3 SMART-board

Rudd (2007) means that teachers find the interactive whiteboard to be motivating for the students. Alexander (2008) and Dawes (2004, 2008) believes that if the students use more dialogue, this becomes significant for their education.

## 2.5 Interest

The teenage years are critical for the adolescent to keep their interest in scientific subjects. The teenager is trying to find their individual goals and identity (Krapp and Prenzel, 2011) and then they often lose their interest in such subjects.

In most cases when interests are introduced from outside we can only keep it alive for a short time, but that does not mean that it is impossible to sustain interest.

In the “Rescue work” it is probable that the students will be working with the robots for the first time, and thus an interest for the subject may be held for a short time. If the situation is stabilized they can keep their interest for a limited time and some students might engage themselves more in the subject.

Research shows that if physics is aimed at presenting scientific nature laws and reconstructing mathematics the students will lose their interest, but if physics lesson is taught in relation to the students’ lives there is a good chance that their interest remains or increases (Krapp and Prenzel, 2011).

### 3 Robotics

A robot is a virtual or mechanical intelligent agent. Robots are often electromechanical machines that are controlled through programming, allowing the robot to make independent movements. A robot can be autonomous or semi-autonomous.

The word Robot can mean both physical robot and virtual software agent, but the latter are typically referred to as bots. It is hard to define the term Robot, but a robot usually can do some or all of the following: move around, operate a mechanical limb, sense and manipulate its environment, exhibit intelligent behavior, or more specifically behavior which mimics humans or other animals. Encyclopedia Britannica (2011-06-29) defines a robot as “any automatically operated machine that replaces human effort, though it may not resemble human beings in appearance or perform functions in a humanlike manner.” Merriam-Webster (2011-06-29) describes a robot as “a machine that looks like a human being and performs various complex acts (as walking or talking) of a human being” or “mechanism guided by automatic controls”. Joseph Engelberger, a pioneer in industrial robotics once said “I can’t define a robot, but I know when I see one.” (Robots-and-robots.com, 2011-06-29).

#### 3.1 History of Robotics

According to Mark E. Rosheim (1994), "The beginning of robots may be traced back to the great Greek engineer Ctesibius (c. 270 BC). Ctesibius applied knowledge of pneumatics and hydraulics to produce the first organ and water clocks with moving figures."

In approximately 1495 Leonardo da Vinci was sketching a plan for a humanoid robot; the robot was meant to be able to sit up, move its arms, head and jaw (Rosheim, 2006).

The term Robot was first introduced by the Czech writer Karel Čapek in his play R.U.R (Rossum’s Universal Robots) (fig 4), published in 1921. The play is about slaves that are called Robots; they do not reflect the modern-day image of a robot. They are machine-similar humans’ not human-similar machines (Tekniskamuseet.se/Robotics, 2011-06-29). It was not Karel himself that coined the word robot, it was his brother Josef. Karel had thought about that he should use the word labori from the Latin word “Labori” which means worker. But his brother then came up with the word “Robota” which in Czech means worker.

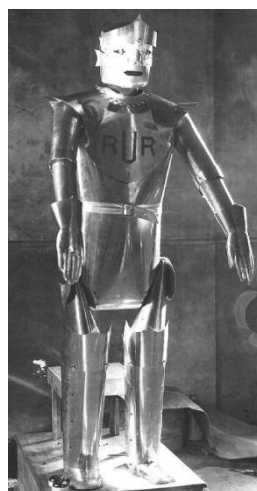


Figure 4 R.U.R. A “robot” from Karel Čapek’s Play “Rossum’s Universal Robots.

### 3.1.1 The three laws of Robotics

The three laws of Robotics were formulated by the writer Isaac Asimov together with John W Campbell in 1942 (Tekniskamuseet.se/Robotics, 2011-06-29).

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey any orders given to it by human beings, except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Later one more law was added:

0. A robot may not harm humanity, or, by inaction, allow humanity to come to harm.

### 3.2 Robotics today

Robotic Research is still today a popular branch, and it is ever progressing. New fields of application are under constant development, for example at the Fukushima Nuclear Power Plant (fig 5) after the nuclear accident in early 2011. Robots have been used to inspect the damage and the clear up process. Robots in medical hospitals, health and service robots, military robots are just some of the new robots that researchers are working with (Tekniskamuseet.se/Robotics, 2011-06-29).



Figure 5 Robot from Fukushima, designed to clean up in the nuclear power plant.

The robots are getting more and more advanced and we are using robots in more fields today than ever before. Sometimes the robots are constructed to replace humans in dangerous situations and sometimes to make processes faster. With robots we can make the precision greater and a robot can work faster than a human. However, not all current robotics projects are actually positive for our society. For example in a current project in Japan, robots are constructed to keep old people company (robotnyheter.se, 2011-08-29). These robots are developed to replace the humans in social situations. In this case an ethical question is raised. This shows that robots are a common item for us in modern society. But we have to discuss how we should and can use them.

### 3.3 Robots in Education

Garcia & Patterson-McNeill (2002) believe that the students have a possibility to control computers with LEGO Mindstorms. After all, it is still possible to learn and have fun at the same time.

Many students have played with LEGO building blocks as children; therefore, they are intrigued with working on LEGO-based classroom exercises. Klassner & Anderson, (2003) mean that constructing robots and design code increases the students' motivation to learn computer science. The use and the programming of real entities brings the students into contact, not only with technology but also with the technical difficulties which generated by the use of real systems creates - a concept which has a very high educational value. It has been showed that Robotics inspires the students in problem solving and team work in computer science (Beer et al., 1999; Hwang et al., 2000).

#### 3.3.1 Lego

Lego bricks can be put together and connected in many ways, to construct such objects as vehicles, buildings and working robots. The possibilities with Lego are countless. You can put them together to create an object, then take it apart again and use the pieces to make another object.

Lego Mindstorms (fig 7) is a programmable construction line from Lego; it was released in July 2006. It consists of a NXT Intelligent Brick (fig 6), sensors, motors and Lego parts. The NXT Intelligent Brick is a 32-bit microcomputer with built-in loudspeaker, 100x64 monochrome LCD display and 4 buttons to navigate a user interface using hierarchical menus. It is powered by 6 AA batteries. The Intelligent brick is the "brain" of a Mindstorms machine; it makes the machine do different operations. The brick can take input from up to four sensors and control up to three motors. It is compatible with both Mac and PC (Mindstorms.lego.com, 2011-06-29).



Figure 6The NXT Intelligent Brick (in the middle) and the three motors, the four standard sensors.



The sensors are of different kinds, for example temperature, light, sound, touch and ultrasonic (Mindstorms.lego.com, 2011-06-29).

LeJOS (LEGO Java Operation System) is a firmware replacement for Lego Mindstorms NXT programmable bricks. LeJOS includes Java virtual machine. It allows the Lego Mindstorms NXT to be programmed in the Java programming language (lejos.sourceforge.net, 2011-06-29).

More information about the classes, functions and how to program with Java and leJOS can be found in Appendix A.1.



Figure 7 Lego Mindstorms. In the center, the body of the robot is the programmable intelligent brick.

## 4 The Results

### 4.1 Description of the “Rescue Work”

Based on the study and observation of the former laboratory works I developed the new laboratory work “Rescue Work” for older students (in upper secondary school). It is the first RoboScope scenario for students of this age and in which they program in Java. It also incorporates a higher degree of freedom and interactive education processes. The “Rescue work” uses some parts from the existing concept used in the RoboScope project (Hansen, Hees and Jeschke, 2010). It has the same structure as the previous laboratory works, with an introduction to robotics, building the robot and programming the robot, but the whole laboratory work “Rescue work” includes more discussion, interactive parts and decision-making. “Rescue work” (‘Rettungseinsatz’) is intended to last for four hours. The idea is that the students should work in groups of two or three; allowing for the students to discuss and make their decisions together as a team. The exercise the robot should complete is to travel inside a nuclear power station, avoid obstacles, localize the leak, and repair it. The “leak” is an infrared source and to solve that problem the robot should have an infrared-searcher and an ultrasonic sensor. “Rescue work” is for students that have already worked with programming and Java, because it would be hard for beginners to learn all about using Java, programming, building a robot and make their own program within the time-schedule. All the students have probably already used some kind of robots in another situation (but it is not necessary for the laboratory work “Rescue work”).

The laboratory work “Rescue work” uses the SMART board with presentations, building instructions, a guide for the assistants, a questionnaire, time schedule and handouts (all that material can be found in Appendix A.1-A.5; the handouts, the guide for the assistants and the questionnaire can be found in both German and Swedish).

The laboratory work “Recue Work” can be divided into different central parts. First the students become acquainted with the machines by playing with robots, both built with Lego Mindstorms and others that can be bought completed (that not necessary are in LEGO). Second they are introduced to robotics and the theme through an interactive presentation about robotics. The students discuss and reflect upon robotics in society today, how they are used. Third the “Rescue work” and the exercise (for the robot to travel inside a nuclear power plant, avoid obstacles, localize the leak and repair the leak) is introduced and discussed with the students, the “real” robot that worked in Fukushima is also described. Three different robots will also be shown for the students to choose between. The students should discuss and argue for one specific robot and build that one with the building instructions. Fourth – the introduction to programming in Java during which the students discuss how to write a code to complete specific tasks with the robots. Here we have the opportunity to write a code together with the students and try the code directly. In the fifth part the students show their work to each other and explain how they have done it (fig 8), what difficulties they have come across and how they have solved them. As a conclusion of the “Rescue Work” the assistants talk about what education and different subjects are required to work with robotics. The last part is the evaluation of the “Rescue Work”; the evaluation is used to see if they have to make some changes to the laboratory works. For helping the assistants and to inspire them to work with the laboratory work a guide was made. In the guidance all information about the laboratory work can be found.



Figure 8 The robot that work in the nuclear power plant, in the front you can also see the obstacles.

#### 4.2 Experience from using “Rescue work”

The material for the “Rescue Work” such as presentations, building instructions, feedback, guidance for the assistants and handouts were developed and improved after discussions and evaluation together with employees and students at IfU.

To try the “Rescue Work” with students I visited a German school. I met a small group of 10 students in year 10 where nine of the students were male and only one of them was female. We only had one double lesson (90 minutes) so we decided to only test the parts introducing programming, the programming-phase and after that they should present their robots and thoughts.

Since we were informed that the students had not had much previous experience in programming we decided to do a longer and more detailed introduction to programming. This was followed by introducing the exercise. After that the students should, in pairs, program their robots. The basic step of programming was undertaken by one of the assistants since it was a part in the older laboratory work in the RoboScope project. Then I did the introduction to Java and it was the same introduction that I had done for the “Rescue work”. The exercise was divided into smaller parts. In the end of this laboratory work I had the opportunity to give the students the feedback-form that I had developed for the laboratory work “Rescue work”. This questionnaire was made using Eljertsson (2005) and Bell (2006) and distributed via the teacher to the students participating in “Rescue work” the day after the laboratory work. In this section I only discuss a few of the questions in the questionnaire which are related to the students’ interest. The answers to the complete questionnaire can be found in appendix B. For the different questions the students could choose a number between 1 and 6, were 1 means that they agree completely and 6 means that they do not agree at all.

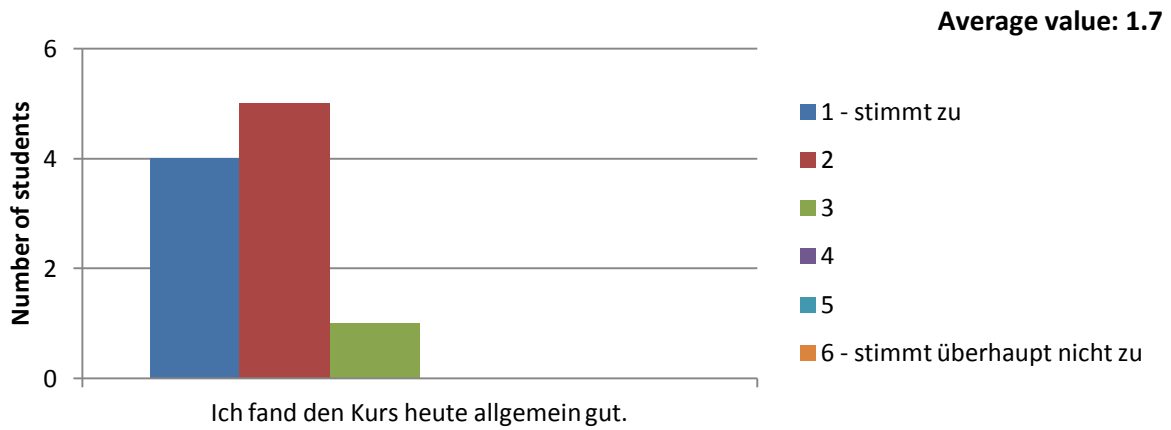


Figure 9 Table with the answers from the question “I found the laboratory work good.” with average value of 1.7.

We only got very preliminary results from the “Rescue Work” since there were only 10 students taking part in the laboratory work. We got some indications about how the students perceived the “Rescue work”. According to the feedback from the students most of them were satisfied with the laboratory work “Rescue work” (fig 9).

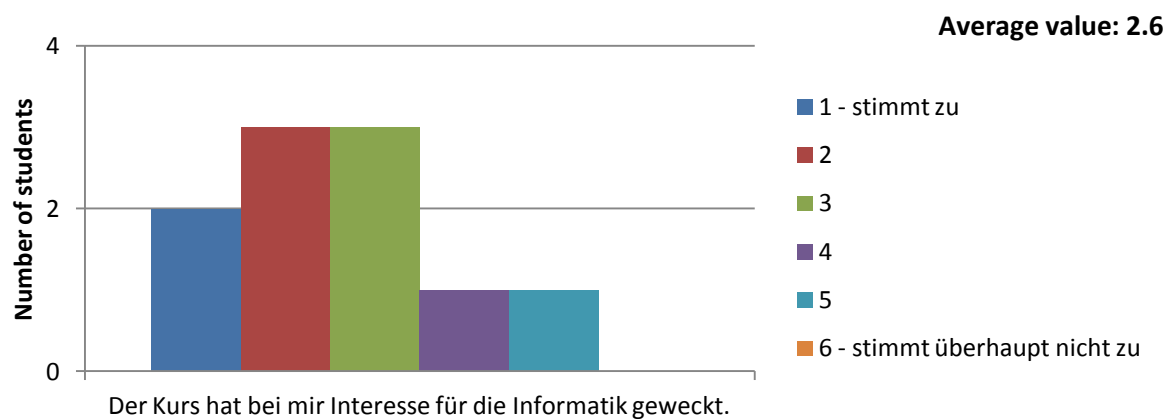


Figure 10 Table with the answer from the question “The laboratory work gave me new interest for computer science.” with average value 2.6.

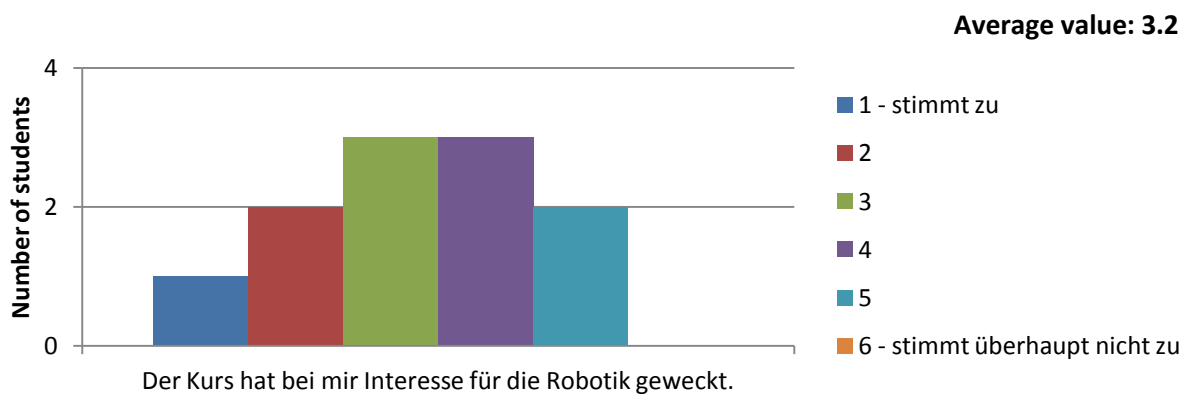


Figure 11 Table over the answers on the question “The laboratory work gave me new interest for robotics.” with average value 3.2

In fig 10 and fig 11 we can see the results from the questions about whether the “Rescue work” succeeded in increasing the students’ interest in the subjects’ computer science and robotics. Those questions do not get high average value (computer science 2.6 and robotics 3.2). We can see that the average value for computer science is a little bit better than for robotics but both of them have a better average value than the middle value (3.5). More focus on programming than on robotics during this particular occasion might explain that the laboratory work resulted in more interest in computer science than in robotics for these students.

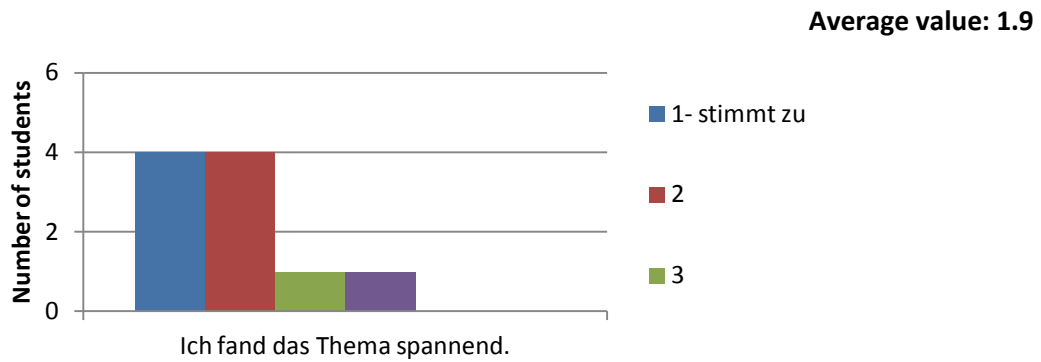


Figure 12 Table with the answers from the question “I found the theme exciting” with average value 1.9.

The question of how exciting the students found the theme got an average value of 1.9 (fig 12). One of the last questions on the questionnaire was what the students think would inspire them to study the technical subjects (fig 13). On this question the students had to formulate their own answers. Practical work and relevance to their lives were the most common answers.

## Was würde bei dir mehr Interesse an den MINT-Fächern (Mathematik, Informatik, Naturwissenschaften, Technik) wecken?

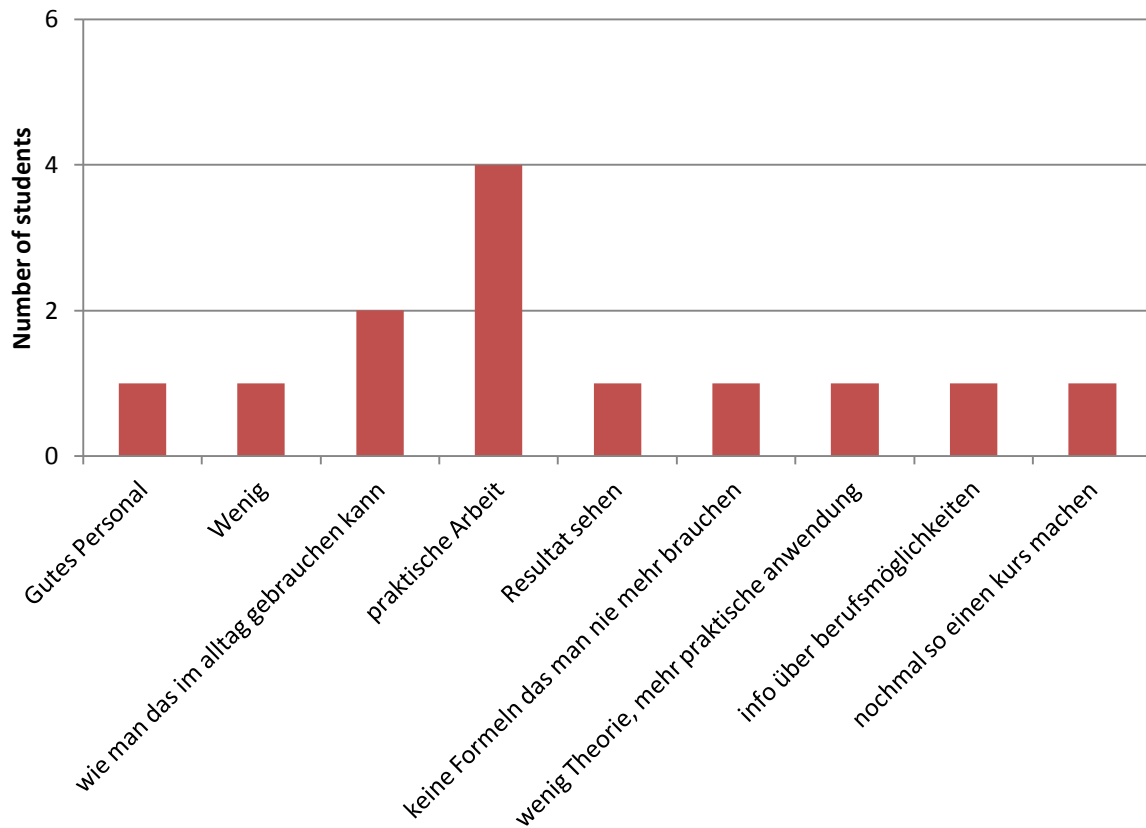


Figure 13 Table over the answers to the question “What would inspire you for the MINT-subjects?” (With the English translation from left to right: Experienced staff, Little, Something that could be used every day, Practical work, To see results, No formula that is never going to be used later, more practical work and not so much theory, information about work possibilities and to do a laboratory work like this again.

## 5 Discussion

My plan with regards to this thesis was to develop a laboratory work for older students using robotics and Java programming based on the model by Brall. This work was to have a higher degree of freedom and more interactivity than previous laboratory works in the RoboScope project. I also wanted to use a current theme that would engage the students and then I came up with the idea that “Rescue work” seemed to fit this criteria. After letting a small group of students try parts of the laboratory work “Rescue Work” and discussing it with colleagues I got a first, preliminary evaluation of the “Rescue Work”. During my work I had to focus on five different points; current theme, interactivity, degrees of freedom, Brall, and interest.

The theme for the laboratory work “Rescue work” was chosen with some thoughts in mind; the theme should be something that the students can relate to (Krapp and Prenzel, 2011), because this can increase the students interest. It was not easy to find a theme that qualifies as an appropriate. One option for a theme was: Japanese nuclear power station. After the big disaster in the beginning of 2011 it had become a well known entity. Nuclear power is a current theme in Germany because in the wake of the Japanese disaster the decision was taken to close all their nuclear power stations. In the laboratory work “Rescue work” the scenario is that we have a nuclear power station that is broken and the nuclear radiation is great. The students should build and program a robot which travels inside the nuclear power station, localizes the leak and repairs it. From the evaluation we can see that the students found the theme exciting (fig 13).

Interactivity was added to this laboratory work and was used in the presentations. From the evaluation we can see that the students found that the active presentations were a good part of the “Rescue Work” (Appendix B: “Welche Dinge findest du am Kurs gut?” or in English: What parts of the laboratory work did you find good?). The results from the evaluation with the school class showed that the assistants did a great job but we could also see that some students found it hard to understand the German language (Appendix B: “Welche Dinge findest du am Kurs gut?” or in English: What parts of the laboratory work did you find good?, “Welche Dinge findest du am Kurs nicht gut?” or in English: What parts of the laboratory work did you not find good? and “Sonstige Anmerkungen” or in English: Other comments).

In the work I tried to increase the degree of freedom. The employees from the institution thought part-solutions (code for each of the different exercises) of the programming would be a good idea. To give part-solutions might be useful if all the students have the possibility to find their own solution and it is not always the same students that get to show their solution and the same students that are given the solution. But it also makes the student lose some of the degrees of freedom; the students have to write the “right” solution, and when we are programming we all have our own ways of writing programs. But I definitely think that it is something that should be tested on one laboratory work “Rescue work”. It might be possible to find a solution that make the students work within their ZPD (Vygotskij, 2001). We also discussed the exercise (move in the nuclear power plant, avoid obstacles, find the leak and repair it) since many of the participants found it hard to understand and unclear, thus wanted to split the exercise in smaller parts and make it more detailed. The problem with that was the ambition to have a high degree of freedom; I wanted the students to think on their own. I had to make the exercise clear so that the students understood what they should do; but so they still have to figure out some things for themselves. The students from the school class thought that trying yourself was one of the good things with the laboratory work (fig 13 and Appendix

B: “Welche Dinge findest du am Kurs gut?” or in English: What parts of the laboratory work did you find good?).

In the “Rescue Work” I used the method from Brall, which also is used in the former laboratory work. The students from the school class thought that the practical work was a positive thing with the laboratory work “Rescue work” (Appendix B: “Welche Dinge findest du am Kurs gut?” or in English: What parts of the laboratory work did you find good?), and we can also see that the students thought that it can make them more interested in the MINT-subjects (fig 13). The laboratory work “Rescue work” is made so that all students have time to work with the exercises but do not necessarily complete all of them, since all the students work at different paces, but together in groups the students can learn more (using ZDP). The students also thought that if the knowledge that they have learned in the laboratory work is useful in their everyday life (fig 13) they could be more interested in the technical subjects. Then the students can relate their experience, do the action and reflect over it, as the model by Brall (2008) mentions.

The aim with the RoboScope project is to engage the students more in the MINT-subjects. The results from questions as to whether the laboratory work “Rescue work” inspires the students in the subjects robotics and computer science can be seen in fig 11 and fig 10. The values were not high but still higher than the middle value so we can assume that we have inspired most of the students for those two subjects. Many of the students have answered that practical work would inspire them for those subjects (fig 13). Some of the other answers about what would inspire the students for those subjects were: more in the daily life, give information about work possibilities and good staff. One student has written that taking part in a laboratory work “Rescue work” again would increase the interest in the MINT-subjects. Even though most of the students found the laboratory work “Rescue work” interesting and enjoyed the work, they did not feel that they would actually need the new knowledge that they acquired. The average on the question as whether they have learned something was 2.0 (Appendix B: “Ich habe etwas gelernt.” or in English: I have learned something) and the average value for if they got some useful knowledge was 3.0 (Appendix B: “Der Kurs hat mir nützliche Kenntnisse vermittelt.” Or in English: The laboratory work has given me some useful knowledge). So they have learned something but many of them did not think that they really will need that new knowledge. During this specific trial session of “Rescue work” the students did not discuss robots and the use of robots, so they could not relate it to something useful. The students only worked with robots seemingly without reason.

In hindsight I have considered my own work a great deal, and if I should restart my work now I think that it would be a good idea to do the tests of the laboratory work “Rescue work” differently and in more detail. It would be good to do a test of the laboratory work “Rescue work” a couple of times with students in the target group, which was the plan from the beginning but it was hard to accomplish. In this manner I could have got better information from the different tests and the feedback to make the laboratory work even better. It is hard to do a laboratory work “Rescue work” with the aim to inspire the students for future studies in the technology subjects and to see if we have made some progress we have to do a test with larger groups.

The laboratory work “Rescue work” gives the assistants at IfU a great possibility to continue working with the development of new laboratory works. This one is, as already mentioned, the first laboratory work done letting the students program in Java and it involves more



interaction than the earlier ones. It is important to be constantly thinking forward and never stop improving the laboratory works.

I think this work also has opened some new windows for future laboratory work within RoboScope since it was done using an interactive whiteboard system. Making laboratory works with a lot of interactive processes can make the students feel that they can influence the work. It makes them feel that they can bring something to the laboratory work. They can learn to describe their thoughts and ideas. They can learn to listen to other students and pay attention to other people. To more actively use the students ideas and thoughts can also be possible in future laboratory works with more discussions. Interactive laboratory works can create more interest for a subject but it can also help the students with their personal development. I have used interactive processes both with the employees and with the school class; the presentations were used to ask questions to the whole group. One example of this is during the “Rescue Work” done with the school class, when I was introducing Java and the exercise but no introduction to robots and robotics, asked what the students thought would happen with the code written on the screen or how I should change the code to make something else with the robots. One of the students answered that this part of the laboratory work “Rescue work” was particular engaging. In the laboratory work with the employees I also used interactive processes in the same way but the difference was that many of them already had experience in the subjects so they already knew the theory. The use of interactive processes gave the participants the possibility to describe what they know and what they have learned. The participants commented that they wanted to involve even more interactive processes (use that even more in the introduction to programming).

It is also possible to make the level of the laboratory work “Rescue work” more difficult. I had the idea from the beginning to do a robot that is driven on all four wheels, but due to the fact that I did not have enough time I had to eliminate that part from the laboratory work “Rescue work” but it would be a great next step. With other constructions the level of the programming can be changed.

For the future it would be interesting to study whether some differences can be found between the genders. It is possible that a better result would be achieved with a separate laboratory work for each gender. Another suggestion would be to continue working with the interactive learning process and to develop and evaluate that part of the laboratory work more thoroughly.

## References

- Alexander, R. (2008). *Toward Dialogic Teaching: Rethinking Classroom Talk* (4<sup>th</sup> ed.). New York: Dialogos.
- Barr, R.B and Tagg, J. (1995). From teaching to learning – A new paradigm for undergraduate education, *Change*, 27, 12-25
- Beer, R., Chiel, H., and Drushel, R. Using Autonomous Robotics to teach science and Engineering. *Communications of the ACM*, June 1999.
- Bell, J. (2006). *Introduktion till forskningsmetodik*. Lund: Studentlitteratur.
- Berg, G. (2003). *Att förstå skolan: En teori om skolan som institution och skolor som organisationer*. Lund: Studentlitteratur.
- Brall, S. (2008). *Arbeitsintegrierte überfachliche Kompetenzentwicklung dargestellt am Beispiel einer technischen Universität*. Inside: Posterpräsentation auf der Tagung „Wagnis Wissenschaft“ – Perspektiven der Promotionsförderung und – Forschung am 5. Und 6. November 2008. Ed. TU Dortmund: Dortmund, 2008.
- Britannica online encyclopedia, (2011-06-29). <http://www.britannica.com/EBchecked/topic/505818/robot>
- Bundesagentur für Arbeit. (2010). *Ingenieurinnen und Ingenieure*. Nürnberg.
- Dawes, L. (2004). Talk and learning in classroom science. *International Journal of Education*, 26(6), 677-695.
- Dawes , L. (2008). Encouraging students' contributions to dialogue during science. *School Science Review*, 90(331), 1-8.
- Douglas, A. Roberts. (1994). Developing the concept of curriculum emphases in science education (inside *Nordisk Pedagogik/Tema: mot en ny didaktikk- naturorienterte emner*).
- Douglas, A. Roberts. (2007). *Knowing Science and Becoming Scientifically Literate: Perspectives on Studying School Science Classrooms*.
- Dysthe, O. (2003). *Dialog, samspel och lärande*. Lund. Studentlitteratur.
- Ejlertsson, G. (2005). *Enkäten i praktiken*. Lund. Studentlitteratur.
- Fraunhofer Institut (2010). *Roberta Programmieren mit Java, Roberts-Reihe Band 1 – NXT*. Stuttgart. Fraunhofer Verlag.
- Fraunhofer Institut (2010). *Roberta Programmieren mit Java, Roberts-Reihe Band 3 – NXT*. Stuttgart. Fraunhofer Verlag.

Garcia, M. A and Patterson-McNeill, H. (2002). Learn how to develop software using the toy Lego Mindstorms. 32<sup>nd</sup> ASEE/IEEE Frontiers in Education Conference, November 6-9, 2002, Boston, MA.

Hansen, A, Hees, F and Jeschke, S. (2010). Hands on Robotics, concept of a student laboratory on the basis of an experience-oriented learning model, Zentrum für Lern- und Wissensmanagement, Institut für Informationsmanagement im Maschinenbau (ZLW/IMA), RWTH Aachen.

Hwang, D.J. and Blandford, D.K. A multidisciplinary team project for electrical engineering, computer engineering, and computer science majors. In Proceedings of the 2000 ASEE Annual Conference (2000).

Klassner, F and Anderson, S. D. (2003). Lego Mindstorms: Not just for K-12 anymore. IEEE Robotics & Automation Magazine, June 2003.

Krapp, A. and Prenzel, M. (2011) Research on Interest in Science: Theories, methods, and findings, International Journal of Science Education, 33:1, 27-50.

Lego Mindstorms NXT. (2011-06-29). <http://mindstorms.lego.com/en-us/Default.aspx>

LeJOS, (2011-06-29). <http://lejos.sourceforge.net/>

Löfdahl, S. (1987). Fysikämnet i svensk realskola och gymnasieskola. (Uppsala Studies in Education 28.) Uppsala, Acta Universitatis Upsaliensis.

Mark E. Rosheim. (1994). Robot Evolution: The Development of Anthrobotics. Wiley-Interscience. New York.

Merriam-Webster Online, (2011-06-29). <http://www.merriam-webster.com/dictionary/robot>

RoboScope. (2011-04-28). <http://www.robo-scope.de/>

Robotnyheter. (2011-08-29). <http://robotnyheter.se/2011/08/25/svt-reportage-om-sociala-robotar-for-aldre-i-japan/>

Robots and Androids, (2011-06-29). <http://www.robots-and-androids.com/what-is-a-robot.html>

Rosheim, E. M. (1994). Robot Evolution: The Development of Anthrobotics. New York: John Wiley & Sons.

Rosheim, E.M (2006). Leonardo's lost robots. Springer Science & Business. New York.

Rudd, T. (2007). Interactive whiteboards in the classroom. Bristol: Futurelab Report: IWBs 2007

Schwab, J. Joseph. (1962). The Teaching of Science as Enquiry. Harvard University Press.

Syh-Jong, J. (2007). A study of students' construction of science knowledge: Talk and writing in a collaborative group. *Educational Research* 49(1), 65-81.

The National Museum of Science and Technology, Exhibition: Robotics, (2011-06-29). <http://www.tekniskamuseet.se/Robotics/index.htm>

Vygotskij, L. (2001). *Tänkande och språk*. Daidalos. Göteborg.

Wikipedia, (2011-06-29). <http://sv.wikipedia.org/wiki/Portal:Huvudsida>

Yager, R. (1995). *Constructivism and the Learning of Science*. I Shawn M. Glynn & Reinders. D. (red.), *Learning science in the school* (p. 35-38). New Jersey: Lawrence Erlbaum Associates, Publishers.

## Figures

Figure 1 RoboScope, logo for the RoboScope project. (2011-04-28). <http://www.roboscope.de/>

Figure 2 Description of the model from Brall, showing the connection between experience, reflection and action. Hansen, A, Hees, F and Jeschke, S. (2010). Hands on Robotics, concept of a student laboratory on the basis of an experience-oriented learning model, Zentrum für Lern- und Wissensmanagement, Institut für Informationsmanagement im Maschinenbau (ZLW/IMA), RWTH Aachen.

Figure 3 SMART-board. (2011-12-28). <http://www.smartboard.se/>

Figure 4 R.U.R. A “robot” from Karel Čapek's Play “Rossum’s Universal Robots. Wikipedia, (2011-06-29). <http://sv.wikipedia.org/wiki/Portal:Huvudsida>

Figure 5 Robot from Fukushima, designed to clean up in the nuclear power plant. (2011-06-29) <http://www.scientificamerican.com/article.cfm?id=robots-arrive-fukushima-nuclear>

Figure 6 The NXT Intelligent Brick (in the middle) and the three motors, the four standard sensors. (2011-12-28). <http://www.robotthoughts.com/>

Figure 7 Lego Mindstorms. In the center, the body of the robot is the programmable intelligent brick. (2011-06-29). <http://mindstorms.lego.com/en-us/Default.aspx>

Figure 8 The robot that work in the nuclear power plant, in the front you can also see the obstacles.

Figure 9 Table with the answers from the question “I found the laboratory work good.” with average value of 1.7.

Figure 10 Table with the answer from the question “The laboratory work gave me new Interest for computer science.” with average value 2.6.

Figure 11 Table over the answers on the question “The laboratory work gave me new Interest for robotics.” with average value 3.2

Figure 12 Table with the answers from the question “I found the theme exciting” with average value 1.9.

Figure 13 Table over the answers to the question “What would inspire you for the MINT-subjects?” (With the English translation from left to right: Experienced stuff, Little, Something that could be used every day, Practical work, To see results, No formula that is never going to be used later, more practical work and not so much theory, information about work possibilities and to do a laboratory work like this again.

## **Appendix**

### **A. Laboratory Material**

## **A.1 Instruktionen für Assistenten / Handledning till assistenterna**

### Schülerlabor Rettungseinsatz

Schülerlabor Rettungseinsatz ist für Schüler in Oberstufe gemacht. Die Schüler sollen schon früher mit Java programmieren können. Diese Labor hat das Thema Rettungseinsatz. Diese Thema sind gewählt weil es in Anfang 2011 ein großer Unfall mit einen Japanisches Kernkraftwerk war, man hat dann versucht mit Roboter in das Kernkraftwerk arbeiten. Das Idee ist dann das alle Schüler früher etwas von das Thema kennen. Das Ziel mit diese Labor ist für zukünftige Studium in die MINT-fächer (Mathematik, Informatik, Naturwissenschaft und Technik) inspirieren, und das Interesse würde größer wenn die Schüler das Thema in Zusammenhang mitbringen können. Mit das Labor wollen wir zeigen dass die MINT-fächer kann Spaß machen und das es gibt viele verschiedenen Möglichkeiten mit diesem Fächer. Dann haben die Schüler auch das Möglichkeit wissenschaftlich Arbeitsprozesse probieren.

Das Labor ist mit einem interaktiven Ausbildungsprozess gemacht und alle Präsentationen sind ganz einfach gemacht. Die sollen zusammen mit dem Schüler komplett gemacht werden. Das Labor benützt die Theorie von Brall; dass bedeutet dass man in einen Kreis mit Reflektion, Erfahrungen und Aktion arbeiten. Das ist wichtig dass die Schüler in die Präsentationen aktivieren werden. Das Diskussion ist auch in diese Labor eine wichtig Prozess, mit das Diskussion haben die Schüler ein Möglichkeit seine Erfahrungen, Reflektion und Aktion zusammenbinden. Die Schüler sollen in diese Labor zwischen drei verschiedene Roboter wählen, jede Team sollen entscheiden zusammen welche Roboter Sie bauen wollen. Dazu sollen Sie auch motivieren warum und wie sie gedacht haben. Dann wollen Sie diese Roboter bauen und das mit Java programmieren. Danach sollen Sie seinen Roboter vorstellen und erzählen wie Sie gedacht haben und was Sie jetzt anderes machen sollen. Dann sollen Sie seinen Erfahrungen mit Aktion und Reflektion zusammenbinden. Die Schüler haben dann auch Möglichkeit seine eigene Entscheid machen, sie entscheiden selbst welche Roboter sie bauen wollen und auch wie Sie das Programmieren sollen. Die Aufgabe (das Ziel) ist schon festgestellt, sie würde ein Roboter bauen. Aber wie sie dann es machen sollen sie selbst entscheiden.

Das Labor anfangen mit einer Begrüßung und dann bekommt die Schüler die Möglichkeit Robotern kennenlernen auf die Demostraße. Danach kommt eine kurze Einführung in Robotik, Präsentation von dem Thema Rettungseinsatz. Dem Schüler bekommt die Möglichkeit zwischen drei verschiedenen Robotern wählen. Die Schüler sollen zusammen in Gruppen diskutieren und sich für einen Roboter entscheiden. Dann sollen die Schüler mit Hilfe von das PowerPoint-Einleitung sein Roboter bauen, eine kurze Einführung von programmieren in Java, danach sollen die Schüler seine Roboter in Java programmieren. Wenn alle Roboter fertig gebaut sind soll die Roboter in die Kernkraftwerk arbeiten und die Schüler haben die Möglichkeit sein Roboter vorstellen. Das Aufgabe ist das wir haben ein Kernkraftwerk mit einen Leck (ein Infrarotquelle) das Roboter soll dann zum Ziel (infrarotquelle), Hindernisse abweichen und die Strahlung messen. Die Schüler können

zusammen diskutieren was Sie gemacht haben, wie Sie gedacht haben und was Sie jetzt anderes machen sollen. Das letzte Teil ist die Ende-Präsentation, Diskussion und Evaluation.

Mit diese freie Materialien sind die Möglichkeiten groß und jede Labor würde etwas Besonderes sein.

## **Das Programm**

- Begrüßung
- Demostraße
- Einführung Robotik und das Aufgabe
- Bauphase
- Einführung Programmierung
- Programmierungsphase
- Vorstellung Roboter
- Ende



## Der Zeitplan

Zeit	Ort	Inhalt	Ziel	Methode	Mittel/Medien	Anmerkungen
0:00-0:05	Eingangsbereich	Begrüßung	-	-	-	Kurze Vorstellung, Jacken/Taschen weg etc.
0:05-0:15	Demostraße	Demostraße	Interesse wecken	Spielen, Anfassen	Demonstratoren	Kurze Erklärungen zu den Demonstratoren/Tischen geben
0:15-0:20	Teachingraum	Teams bilden, Organisatorisches	-	-	-	3-er oder 2-er Teams
0:20-0:35	Teachingraum	Einführung in die Robotik	allg. Info zur Robotik	SMART	Robotik.netbook	Schüler mit einbeziehen, je nach Bedarf mehr oder weniger Details.
0:35-0:40	Teachingraum	Aufgabenstellung	Ziel für Kurs setzen	SMART	Kernkraftwerk. Netbook	Präsentation deutlich machen, Hilfen alle Teams anfangen mit die Diskussion
0:40-1:25	ExperimentierTische	Roboter zusammenbauen, Diskutieren und zwischen die Modulen wählen	Einen Roboter gebastelt haben.	Diskussion. Bauen nach Anleitung	PPT auf den Laptops (Konstruktion1-4.ppt) + Kästen + Fragen	Am Ende die schnelleren selber dekorieren lassen. Über Schultern schauen, Fehler beheben.
1:25-1:40	Teachingraum oder Exp.	Einführung in Java & LeJOS	Java-software und LeJOS-Paket kennenlernen, Parallelen zum Vorherigen ziehen!	Interaktiv, in Java zeigen und gleichzeitig ausprobieren lassen	SMART-Board, Java, eventuell Hand-Zettel, java.netbook	Sehr kritische Phase, erst weitermachen wenn alle Teilnehmer verstanden haben.
1:40-3:15	ExperimentierTische	Selbständiges Programmieren	Selbständiges Experimentieren und Anwenden von gelernten Inhalten	Selber ausprobieren, und aus eigenen Fehler lernen.	Java, USB-Kabel, gebauten Roboter, Zettel, Laptop	Fragen so schnell wie möglich beantworten, außerdem Motivieren! Teilnehmer sind teilweise zu schüchtern zum Fragen!
3:15-3:35	Arena	Vorstellung der Roboter und Diskussion.	Ein Ziel haben, die Motivation erhöhen, einen guten Abschluss bieten	Die Schüler vorführen lassen, ggf. moderieren	Arena Tisch, fertige und programmierte Roboter	Alle zum Vorführen motivieren und diskutieren was sie anderes jetzt machen soll und so weiter (Fortsetzung beim Schluss)
3:35-3:45	ExperimentierTische	Roboter auseinander bauen, währenddessen Website+ Gästebuch vorstellen	Information von Website weiterleiten	SMART	Website	Auch auf andere Kurse hinweisen
3:45-3:55	ExperimentierTische	Schlusswort	Guten Eindruck hinterlassen	SMART	Ende.netbook	Teilnehmern sowie Begleitern möglichst zum Reklame-Machen anregen.
3:55-4:00	ExperimentierTische	Feedback	Feedback zurückkriegen	Fragebogen	Feedback-bögen	Vor allem zur konstruktiven Kritik aufrufen, erklären: "auch negative Kritik ist gute Kritik"!

## Bevor das Labor

- Den Hallen-Schlüssel leihen (mit WC-chip)
- 1x USB-Stick (mit Alle Präsentationen auf)
- Eventuell der Fotokamera oder der Video-Cam ausleihen
- Teilnehmerliste bereit haben
- Drucken die Handouts (für Java)
- Die Halle vorbereiten
  - Auf Sauberkeit achten (eventuelles Aufräumen)
  - Eventuell Heizung aufdrehen
  - Sitzsäcke geschickt positionieren
  - Mülleimer/Mülltüte aufstellen
  - Demo-Straße vorbereiten
    - Tische aufstellen
    - Batterien/Akkus kontrollieren
    - Roboter positionieren (eventuell mit Hilfe von Kursmappe?)
    - Fernbedienungen zuordnen
    - Anleitungs-Zettel zuordnen
  - Experimentier-Arena vorbereiten
    - Tische richtig stellen
    - Genügend Stühle aufstellen (möglichst nicht bei Leinwand)
    - Mehrfach-Steckdosen geschickt verteilen
    - Laptops und NXT-Kästen verteilen (**Je 1x Laptop, 1x weißer Kasten und 1x blauer Kasten pro Team**)
    - USB-Kabel verteilen
    - SMART-board mit Präsentations-Laptop testen, alle notwendige Präsentationen vorbereiten und die Website hinstecken.
  - Teaching-Raum vorbereiten
    - Tische richtig stellen
    - Genügend Stühle aufstellen
    - SMART-Board mit Präsentations-Laptop testen, alle notwendige Präsentationen vorbereiten und vielleicht die Website hinstecken.
  - Wettkampf-Arena vorbereiten
    - Dekorierung dem Szenario entsprechend
    - Die Kernkraftwerke vorbereiten
  - NXT-Kästen vorbereiten
    - Nachgucken so alle Teile gibt in die Kästen
    - NXT's auf Ladung kontrollieren
    - Programm-Speicher leermachen
    - Eventuell Kalibrierungsprogramm auf die NXT's laden
    - Nachgucken so dass die NXT's hat LeJOS installiert
  - Die Laptops vorbereiten
    - Jeder Laptop soll eine Maus und ein Ladegerät haben
    - Java und LeJoS-Software installiert?
    - Alle Konstruktionsanleitungen auf alle Laptops tun
    - Internet-Einstellungen?
      - Die NXT-Blöcke vorbereiten

- Auf alle Blöcke lejos installieren (benutz den nxjflash das soll auf dem Schreibtisch von jeder Laptop sein)Installation: verbinden das Block mit der Laptop, öffnen nxjflash und schreib einfach 1 wenn das Programm nach etwas fragen.
- Feedback
  - Fragebögen drucken

# Detailliert Laborprogramm

## 1. Begrüßung

- Alle Schüler willkommen und ein bisschen von dem Labor des Tages erzählen.

## 2. Demostraße

- Die Schüler haben die Möglichkeit selbst die verschiedene Roboter probieren

## 3. Teams organisieren

- 2er Oder 3er Teams organisieren

## 4. Einführung in die Robotik

- Präsentation mit Hilfe von das Dokument *Einführung Robotik* halten
- Detailliert Beschreibung von das Präsentation:
  - **Seite 1: Robotik.** Was ist denn Robotik? Hat jemand ein Vorschlag? (hier können wir neu Dingen auf das Präsentation schreiben) Lass ein bisschen Zeit und warten auf die Antworten von die Schüler.
  - **Seite 2: Was ist eigentlich ein Roboter?** Unter den Text ist slaw. : robota=arbeiten gesteckt.  
Wir sollen ihr ein Roboter beschreiben? Die meisten von uns kennen Roboter aus vielen verschiedenen möglichen Bereichen des Lebens. Wir verwenden den Begriff ganz selbstverständlich, dabei wissen die meisten von uns wahrscheinlich gar nicht genau, woher er kommt und was er bedeutet.  
Hat jemand von euch eine Idee, was „Roboter“ heißen könnte? (Assoziationen aufgreifen)  
Slawisches Wort: „arbeiten“ (im Sinne von körperlicher Arbeit)  
In seiner heutigen Bedeutung wurde der Begriff erstmals 1920 in einem tschechischen Theaterstück verwendet, in dem es um menschenähnliche Maschinen geht, die hergestellt werden, um den Menschen die Arbeit zu erleichtern.
  - **Seite 3: Bestandteile eines Roboter und ihre Gegenstücke**  
Was ist ein Roboter? Eine Maschine, die dem Menschen Arbeit abnimmt. Doch was heißt das genau und wie macht er das eigentlich? Es gibt so viele Beispiele für und Arten von Robotern, doch sie alle haben gemeinsam, was man so zusammenfassen kann: Ein Roboter ist etwas, das Interaktion mit der physischen Welt auf Basis von Sensoren, Aktoren und Informationsverarbeitung umsetzt.  
Vergleichen mit dem Menschen(Sensoren/Sinne, Aktoren/Muskeln, Gehirn/NXT-Block)
  - **Seite 4:Interaktion.** Aber was ist eine Interaktion? Kann jede Art von Wechselwirkung mit der Umgebung sein! ebenso wie bei uns Menschen: Sprechen, Hören, Fühlen, Greifen, Gehen...Mechanisch: Greifen etc. Physikalisch: Temp. Etc.Kummikativ: Signale empfangen/losschicken etc.
  - **Seite 5: Roboter sind besondere Maschinen.** Roboter sind stationäre oder mobile Maschinen, die nach einem bestimmten Programm festgelegte Aufgaben erfüllen, aber in Vergleich mit einer Maschine kann man das Programm verändern.
  - **Seite 6: Robotergesetze**  
0. Ein Roboter darf die Menschheit nicht verletzen oder durch Passivität zulassen, dass die Menschheit zu Schaden kommt.  
1. Ein Roboter darf keinen Menschen verletzen oder durch Untätigkeit zu Schaden kommen lassen, außer er verstieße damit gegen das nullte Gesetz.

2. Ein Roboter muss den Befehlen der Menschen gehorchen – es sei denn, solche Befehle stehen im Widerspruch zum nullten oder ersten Gesetz.
3. Ein Roboter muss seine eigene Existenz schützen, so lange dieses sein Handeln nicht dem nullten, ersten oder zweiten Gesetz widerspricht.

Zuerst war es nur drei verschiedene Gesetze aber später in 1986 wurde noch ein Gesetz hinzugefügt. Von Isaac Asimov zusammen mit John W Campbell kurz nach 1940 gemacht.

- **Seite 7: Robotarten.**

In jede Ecke gibt es einer Art gesteckt. Oben links: Humanoide, Unter links: Spielzeug, Oben rechts: Industrie, Unter rechts: Erkundung (Militär).

Was für Roboter gibt es denn so heute? Keine Vorschlag auf der Präsentation aber kann die Schülern nicht Arten finden so gibt diese: Humanoide (die Roboter sind möglichst menschenähnlich, und sehr kompliziert (erst seit in den letzten Jahrzehnten verwirklicht)), Industrie (die z.B. bei so gut wie jedem Autohersteller eingesetzt werden. Diese übernehmen Arbeiten, die sie schneller und/oder genauer als Menschen machen können. z.B. Schweißen, Halten, Nieten, Löten, usw.), Service ((dazu auch Spielzeug) Serviceroboter sind Roboter die z.B. Haushaltsarbeiten wie Staubsaugen oder Rasenmähen erledigen. Ein ganz besonderer Roboter ist der sog. „Care-O-bot“. Dieser ist ein Pflegeroboter, der kranken, alten oder behinderten Menschen hilft im alltäglichen Leben. Zu den Servicerobotern gehören allerdings auch die Spielzeugroboter. Bsp.: „Aibo“ von Sony, ein künstliches Haustier das läuft, liegt, sitzt und mit einem Ball spielen kann.), Erkundung((Militär) Erkundungsroboter werden u.a. vom Militär, Polizei, und Raumfahrtgesellschaften eingesetzt. Diese haben meistens eine Aufklärungs-Funktion, oder übernehmen Aufgaben die für einen Menschen zu gefährlich wären (z.B. Rettungseinsätze, Bombenentschärfung etc.) Sehr bekannt: die NASA Mars-Rover, „Big Dog“, ein mechanischer Packesel für Militärische Zwecke, eine Drone (unbemanntes Flugzeug) der amerikanischen Luftwaffe, Ein Erkundungsroboter sucht nach Überlebenden nach einem Bergbau-Unfall, Ein Minenräumungsroboter. Man sieht also: Roboter sind unglaublich vielfältig einsetzbar! Versteckende Fotos das Mann zeigen kann.

Klar würden alle diese Roboter in eine Labor kennengelernt, das ist genug nur ein paar nennen. Aber das ist ja gut ganz viele kennen, weil ihr die Schüler fragen. Und dann wissen ihr bevor nicht welche Antwort ihr bekommt.

- **Seite 8: Lego Mindstorms.**

Und zwar wollen wir ein Baukasten-System benutzen, genauer gesagt den LEGO Mindstorms NXT Baukasten. In diesem System finden wir wirklich alles was wir brauchen um eigene Roboter zu bauen und programmieren. Nämlich Sensoren... Zum Beispiel: Drucksensoren, Geräuschsensor, Lichtsensor, Ultraschallsensor und Magneten Sensor. Aktoren: 3 Motoren, Display, Lautsprecher und Lämpchen) und eine programmierbare Einheit zur Informationsverarbeitung („Gehirn“)!

## 5. Rettungseinsatz-Einführung

- Präsentation von die Aufgabe mit Hilfe von das Dokument *Rettungseinsatz*
- Detailliert Beschreibung von die Aufgabe

- **Seite 1: Rettungseinsatz.**

Willkommen beim RoboScope-Kurs „Rettungseinsatz“. Das gibt viele verschiedene Aufgaben für einen Rettungsroboter.

- **Seite 2: Fotos von Fukushima**

Zum Beispiel kann der Roboter in ein Kernkraftwerk arbeiten. Nach dem Unfall in Japan Anfang 2011 haben Roboter drinnen das Kernkraftwerk gearbeitet.

- **Seite 3: Fotos von Roboter das würden drinnen Fukushima.** Hier können wir Fotos sehen von dem Roboter das sollte in Fukushima arbeiten.
- **Seite 4: Noch ein paar Fotos von Roboter in Action drinnen Fukushima**
- **Seite 5: Unser Roboter soll:** den Strahlungsquelle finden (infrarotquelle), Navigieren, Die Strahlung messen und zur Strahlungsquelle fahren. Heute sollen ihr in 2er oder 3er Teams ein Roboter basteln mit Lego Mindstorms und danach das Robot in Java programmieren. Der Roboter hat ein paar verschiedene Aufgaben dass es machen soll.  
Achtung! Die verschiedenen Aufgaben steht nicht in der Präsentation in Arbeitsordnung!
- **Seite 6: Dazu brauchen wir** verschiedene Sensoren (InfrarotSensor, Ultraschallsensor). „ Motoren (so wir können der Robot fahren). Und der NXT-Block (So wir können das Robot programmieren)
- **Seite 7: Die verschiedenen Wahlmöglichkeiten schnell präsentieren.** An was müssen wir denken wenn wir einen Roboter bauen? Stabilität, Zugänglichkeit (Sensoren), Konstruktionsgedanken, Ästhetisch angenehm. Diskutieren zusammen und versuchen zu erklären und beschreiben ihre Wähl. Warum macht ihr so? Vorteilen? Nachteilen? Die verschiedene Modulen beschreibe und erzählen das sie sollen selbst wählen und motivieren warum ihr hat diese Modul gewählt. Erzählen von das Moment wenn sie sollen seine Roboter vorstellen. Die Schüler sollen versuchen die ganze Zeit kritisch denken und alles motivieren.
- **Seite 8: Jetzt geht's los!**

## 6. Diskussion und Bauphase

- Die Schüler Anfängen mit ein Diskussion welche Modul die sollen bauen und warum. Vielleicht brauchen die Schüler Hilfe von den Assistenten die Diskussionen anfangen. Können gern ein bisschen Notizen machen.
- Wenn die Schüler haben gewählt welche Modul sie bauen sollen ist es nur zu anfangen. Dazu auch die Sensoren an die Module bauen. Die Bauinstruktionen gibt für jede Konstruktion: *Konstruktion1* bis *Konstruktion3* in PowerPoint.

## 7. Einführung in die Java und LeJOS (vielleicht auch möglich mit NXT-G)

- Präsentation mit Hilfe von das Dokument *Einführung Java*
- Detailliert Beschreibung von das Präsentation
  - **Seite 1: Einführung in Java**  
Heute sollen wir in Java programmieren. Ich glaube alle von euch schon früher mit Java programmiert haben. Aber das gibt ein paar neue Dingen das wir lernen müssen wenn wir wollen das NXT programmieren.  
Was ist eigentlich Programmierung? Was denkt ihr? (Möglichkeit etwas schreiben auf die Präsentation.  
Jeder hat zumindest eine große Vorstellung, was unter „Programmierung“ zu verstehen ist, heute wollen wir es aber mal richtig erklären. Wichtige Schlagwörter, auf die man auch hinarbeiten kann: Befehle, Ablauf/Reihenfolge, Selbstständig/Automatisch, Denken/Intelligenz
  - **Seite 2:**  
Computer sind dumm  
*Weil: sie nur ganz kleine Schritte verstehen können.*  
Computer sind faul  
*Weil: sie nichts tun, wenn man es ihnen nicht vorher sagt.*

Computer brauchen einfache Befehle hintereinander. Programmierung beinhaltet instruieren oder erzählen für den Maschinen oder Roboter was zu tun. Das kann man in viele verschiedene Programmiersprachen machen, heute werden wir das in den objektorientierende Programmiersprache Java machen. Das bedeutet das man muss in sehr kleine Schritte programmieren und das ganze Programm sehr deutlich und organisiert machen. Wir kann ein Mensch sagen tue das, aber das wird kein Roboter verstehen. Dann muss wir jede kleine Schritte erklären, zuerst umdrehen, dann Forwards gehen... Java ist ein objektorientiert Programmiersprache (ein Satz Objekt das mit einander interagieren). Wenn wir wollen unser Motoren und Sensoren benutzen braucht Java das LeJOS-Paket, so dass es mit NXT kommunizieren kann. (leJOS = LEGO Java Operation System). Wir müssen dann lejos.nxt.\* importieren: Import lejos.nxt.\*

- **Nächste Schritt wäre dann Eclipse öffnen.** Zeigen wir man ein Programm machen und wie man die verschiedene Funktionen benutzen. Mach das interaktiv zusammen mit die Schülern, das bedeutet dass wir wissen jetzt nicht wie das aussieht. Aber hier gibt trotzdem ein Beispiel:

```
import lejos.nxt.*;
import lejos.nxt.addon.IRSeekerV2;
import lejos.nxt.addon.IRSeekerV2.Mode;
public class Test {
    public static void main(String [] args) throws Exception{
        while (!Button.ESCAPE.isPressed()){
            IRSeekerV2 Sucher = new IRSeekerV2(SensorPort.S2, Mode.AC);
            //Erstellt den Infrarotsensor mit dem Namen Sucher.
            UltrasonicSensor sonic = new UltrasonicSensor(SensorPort.S2);
            Sucher.setAddress(0x10);
            //Notwendige Einstellung für den Sensor.
            int direction = Sucher.getDirection();
            int distanz = sonic.getDistance();
            if (distanz > 20){
                Motor.A.forward();
                Motor.B.forward();
                Thread.sleep(1000);
            }
            else{
                LCD.drawString(direction,0,1);
            }
        }
    }
}
```

**Schleifen:** Eine Schleife wiederholt einen Anweisung-Block so lange, wie eine Laufbedingung gültig ist oder bis eine Abbruchbedingung eintritt. For-Schleife: die Zählvariable wird in innerhalb der for-Schleife initialisiert, danach folgt die Bedingung, abschließend bestimmt der sogenannte Fortschaltausdruck, wie mit der Zählvariablen umzugehen ist. For (int x = 8; x >= 0; x-). If-else Anweisung: if wird verwendet, um Programmverzweigung aufgrund einer Bedingung zu realisieren. Nur wenn die Bedingung zutrifft, werden die Anweisungen nach dem Schlüsselwort if abgearbeitet, ansonsten werden die Anweisungen nach else abgearbeitet.

**leJOS.** Es gibt viele verschiedene Klassen in das LeJOS-Paket, aber wir brauchen heute nicht alle. Das gibt viele verschiedene Klassen, zum Beispiel: Lightsensor, Ultrasonic Sensor, Motor, LCD und TouchSensor.

**Display.** Auf diesem können sowohl textnachrichten als auch Messwerte oder Sensorwerte ausgegeben werden. Beim Display des NXs handelt es sich um ein LC-Display mit den Maßen 100 \* 64 Pixel. Ein Zeichen hat die Höhe 8px und die Breite 6px, somit können in einer Zeile 16 Zeichen untergebracht werden. Durch die Zeichenhöhe ergibt sich, dass maximal acht Zeilen zur Verfügung stehen, beginnend bei Zeile null bis Zeile sieben. Der Ursprung des Koordinatensystems befindet sich, wie in Java üblich, am oberen linken Display.

Die Klasse LCD:

DrawString(string text, int x, int y)

Clear()

Scroll()

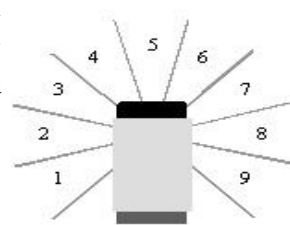
**Sensoren.** Die Sensoren des NXT sind seine „Augen und Ohren“, über sie kann der Roboter Informationen über seine Umwelt sammeln. Im Standard-Paket sind vier Sensoren enthalten: Ein Lichtsensor, ein Ultraschallsensor, ein Geräuschsensor und ein Berührungssensor. Während der Berührungssensor nur erfassen kann, ob er gedrückt ist oder nicht, liefern die anderen Sensoren einen quantitativen Wert zurück.

**Ultraschallsensor.** Der Ultraschallsensor misst den Abstand zu Objekten in cm. In einem Winkel von ca. 30° (-15° bis + 15°) werden vornehmlich eckige Gegenstände mit einer glatten Oberfläche erkannt. Die maximale Entfernung dabei entspricht ca. 250 cm- Sobald die Ultraschallwellen nicht frontal auf den Gegenstand treffen, ergeben sich sehr schnell große Messfehler. Bei einer Entfernung von 150 cm sollte das zu erkennende Objekt nicht weiter als 20° vom Winkel des Ultraschallsensors gedreht sein. Verringert sich die Entfernung zwischen Gegenstand und Sensor, werden auch stärker gedrehte Objekte erkannt.

Die Klasse UltrasonicSensor:

int getDistance()

**Der Infrarotsensor.** Der Infrarotsensor misst die Strahlung von einem Infrarot Quelle. Das IRSeeker benutzt ein digital Signal und filtern die empfängt Signalen. Mit der IRSeeker können wir ein Wert (siehe das Bild unten) oder Graden (zwischen -180 und 180) als Rückwerte bekommen.



Die Klasse IRSeekerV2:

getAngle()

getDirection

getSensorValue()

**NXT-Motor.** Die Motoren des NXT haben eine maximale Winkelgeschwindigkeit von 900° pro Sekunde, was in etwa 2,5 Umdrehungen pro Sekunde entspricht. Die Motoren des NXTs können in 1°-Schritten angesteuert werden. Dem NXT Motor wird nicht die zu erreichende Geschwindigkeit vorgegeben, sondern die aufzuwendende Kraft. Da die Motoren Servomotoren sind, verfügen sie über interne Rotationssensoren, die messen, um wie viele Grad sie gedreht wurden. Auf diese Weise kann zum Beispiel ein zurückgelegter Weg berechnet werden. Somit können die Motoren sowohl als Aktoren als auch als Sensoren genutzt werden.

Die Klasse Motor:

setSpeed(int gps)

setPower(int kraft)

forward()

backward()



```
stop()
rotate(int winkel)
int getTachoCount()
shutdown()
```

- Öffnen das Dokument **Aufgabe. Unser Roboter soll können:** Bewegung, vorwärtsfahren/rückwärtsfahren, drehen um  $x^\circ$  Graden, die Strahlungsquelle finden, Bewegung zum Ziel und Hindernisse abweichen. Diskutieren die Aufgabe und was wir sollen mit jede Sensor und Motor machen. Das wäre gut hier anfangen ein bisschen Tipp geben und kreativ sein, so dass die Schüler selbst später denken können. Zusammen ein bisschen diskutieren was die verschiedenen Teilaufgaben bedeutet.

## 8. Programmierungsphasen

- Die Schüler soll in diesem Teil selbstständig sein Roboter programmieren. Die Schüler haben Hilfe von dem Java-Zettel. Die Assistenten sind hier ein Hilfe für die Schüler wenn sie mag.
- Versuch keine Antwort geben sonst nur ein Anfang in die Gedanken geben.
- Die Handouts ausgeben

## 9. Wettkampf-Arena

- Alle Team zeigen seine Roboter und sollen auch erklären warum die haben so gemacht. Und was sie haben gedacht. Vielleicht muss ihr hier anfangen Fragen stellen, so das Diskussion gut anfangen kann. Zum Beispiel kann ihr Fragen: Was sollen ihr jetzt anderes machen?

## 10. Die Roboter löschen. Gleichzeitig die Website introduzieren.

- Deutlich sein das es gibt ein Sortierung in die Kästen und sollen danach auch so sein, weil das wäre einfacher für nächste Teilnehmer.

## 11. Ende

- Das Dokument **Ende** benutzen
- Diskussionen, Fragen und fortschreitende Studium
  - **Seite 1: Zum Schluss.**
  - **Seite 2: Diskussion und Fotos:** Was haben wir gelernt? Was sollte ihr jetzt anderes machen? Warum? Resultat. Benutz die Möglichkeit auf dem SMART-board schreiben. (Hier muss man in jeder Diskussion die guten Fragen finden, jede Diskussion ist speziell).
  - **Seite 3: Beruf: Roboter bauen.** Frage zu die Schüler: Was denkt ihr jetzt das man braucht wenn man ein Roboter bauen? Welche Fächer sollen man studiert hat? Welche hat die richtige Beruf ein Roboter bauen?
  - **Seite 4: In Robotern steckt...** ... Informatik... Mathematik... Physik... Chemie ... Biologie/Biotechnologie... Elektrotechnik... Maschinen- und Verkehrstechnik... Mechatronik... Energietechnik... Werkstoffwissenschaften... Technikgestaltung/Design
  - **Seite 5: Aktuelle Projekten** zum Beispiel: Medizinische Roboter, Care-O-bot, Elektronische „schienen“ für LKW, Bewegungsroboter die sich in der Nähe von Menschen bewegen. (Hier können ihr ganz einfach das Präsentation verändern, das wird besser ob das sind Projekten das ihr gefällt.) Dieser Teil wird noch ein Möglichkeit die Schüler in das Thema interessieren.
  - **Seite 6: Fragen?**
  - **Seite 7: Feedback antworten**

## **Tipp!**

- Geschlossene Laptops während programmierung-Vortrag

## **Naher das Labor**

- Die Halle aufräumen
  - Müll, Kästen, Laptops
  - Alle Demo-Roboter ausschalten
- Die Kamera zurückgeben
- Die USB-stick zurückgeben
- Die Feedback durchlesen und zusammenfassen.
- Die Schlüssel zurückgeben

## Skollaborationen Räddningsinsats

Skollaborationen Räddningsinsats är skapad för elever i Oberstufe (den högsta delen va det tyska gymnasium, dvs. 10-12 klass). Eleverna ska redan kunna programmera med java. Laborationen har temat räddningsinsats. Temat är valt eftersom det i början av år 2011 var en stor olycka med ett japanskt kärnkraftverk, då man försökt att arbeta med robotar i kärnkraftverket. Temat är då något som förhoppningsvis alla elever tidigare hört talas om. Målet med denna laboration är att inspirera till framtida studier inom MINT-ämnena (Matematik, Datakunskap, Naturvetenskap och Teknik), och förhoppningsvis blir intresset större när eleverna kan sätta temat i något sammanhang. Med denna laboration vill visa eleverna att de MINT-ämnena även kan innebära roliga aktiviteter och att det finns många möjligheter med dessa ämnen. Då har eleverna också möjligheten att testa arbetsprocessen.

Denna laboration använder interaktiva inlärningsprocesser och alla presentationer är enkla. Presentationer ska sedan kompletteras tillsammans med eleverna. Laborationen använder teorin av Brall, dvs. att man arbetar i en cirkel med reflektion, erfarenheter och aktivitet. Det är viktigt att eleverna är aktiva under presentationerna, även diskussion är en viktig del i denna laboration, med diskussionen har eleverna möjlighet att binda samman erfarenheter, reflektioner och aktivitet. Eleverna har möjligheten att välja mellan tre olika robotar, varje grupp ska tillsammans bestämma vilken robot de vill bygga. Eleverna ska även motivera varför de valt just den robot de valde. De ska sedan programmera sin robot i java. Sedan visa sin robot för de andra grupperna när den försöker genomföra den uppgift de fått. De kan då även diskutera mellan grupperna om sina olika val och vad de tycker om resultatet. Uppgiften är alltså att eleverna ska bygga en robot (och programmera den), men vilken robot de ska bygga och hur de ska programmera den är upp till eleverna att bestämma.

Laborationen börjar med att eleverna hälsas välkomna och sedan får de möjlighet att bekanta sig med de robotar som ställts fram, det är en blandning mellan olika robotar som har en specifik (t.ex. en robot-hund, en dans-robot) och robotar som är byggda med Lego Mindstorms (vilket gör att uppgifterna kan varieras). Sedan är det dags för en introduktion om robotik, presentation om temat: räddningsinsats. Därefter börjar byggandet av robotarna, där eleverna kan välja mellan tre olika robotar, när de sedan har hjälp av en PowerPoint för att bygga sin robot. När robotarna är byggda är det dags för en kort introduktion till java och hur man använder java för att programmera lego robotarna. Robotarna programmeras sedan innan det är dags för eleverna att visa sina kompisar vad de har lyckats med och hur de tänkte när de arbetade med roboten. Uppgiften som de ska programmera sin robot för att genomföra är att i ett kärnkraftverk hitta den läckan (dvs. lokalisera och köra till läckan som är en infrarödkälla), på vägen till läckan ska roboten kunna undvika hinder och den ska även kunna mäta strålningen som kommer från källan. Avslutningsvis blir det en kortare presentation, diskussion och utvärdering.

Med detta fria material är möjligheterna stora och varje laboration kan utvecklas till något speciellt.

## **Program**

- Hälsning
- Demoyta
- Introduktion till robotik och uppgiften
- Byggdel
- Introduktion till programmering
- Programmeringsdel
- Framställning Robotarna
- Avslutning

## Tidsplan

Tid	Plats	Innehåll	Mål	Metod	Hjälpmedel/Media	Kommentarer
0:00-0:05	Ingången	Hälsning	-	-	-	Kort framställning, lägga bort jackor, väskor osv.
0:05-0:15	Demoyta	Demoyta	Väcka intresse	Leka	Demonstration	Ge korta förklaringar om robotarna
0:15-0:20	„Klassrummet“	Skapa grupper, organisatoriska uppg.	-	-	-	Grupper om 2 eller 3.
0:20-0:35	„Klasrummet“	Introduktion till robotik	Allmän info om robotik	SMART	Robotik.netbook	Inkludera eleverna, mer eller mindre detaljer beroende på behov.
0:35-0:40	„Klassrummet“	Uppgiftsintro.	Sätta mål för laborationen	SMART	Kernkraftwerk. Netbook	Göra presentationen tydlig. Se till att alla grupper kommer igång med diskussionerna.
0:40-1:25	ExperimentYta	Bygga robotar, diskutera och välja mellan de olika robotarna.	Byggt en egen robot.	Diskussion. Bygga enligt beskrivning.	PPT på datorerna (Konstruktion1-4.ppt) + Lådor + Frågor	Låt de snabba själv dekorera sina robotar. Hålla koll på grupperna och se om de behöver hjälp.
1:25-1:40	„Klassrummet“ eller Exp. Yta	Introduktion till Java & LeJOS	Lära känna Java-mjukvara och LeJOS-paketet. Dra paralleller till tidigare erfarenheter.	Interaktivt, visa i Java och samtidigt testa.	SMART-tavla, Java, handouts och java.netbook	Mycket kritisk fas, gå inte vidare förrän alla har förstått.
1:40-3:15	ExperimentYta	Självständigt programmera	Självständigt experimentera och använda införskaffade kunskaper	Själva pröva och lär av egna misstag.	Java, USB-Kabel, byggda robotar, handout och dator.	Frågor så snabbt så möjligt besvara och motivera eleverna! Deltagarna är ibland för blyga för att fråga själva.
3:15-3:35	Arena	Framställning av robotarna och diskussion.	Ha som mål att höja motivationen, och ge ett bra avslut.	Låta eleverna visa sina robotar.	Arenayta, färdiga och programmerade robotar.	Motivera alla till att visa sina robotar och diskutera kring vad de skulle gjort annorlunda osv. (Fortsätta under resterande av programmet.)
3:35-3:45	ExperimentYta	Bygga isär robotar, lisa webbsidan och gästboken.	Ge information om hemsidan.	SMART	Hemsidan	Också prata om andra laborationer.
3:45-3:55	ExperimentYta	Slutord	Lämna ett bra intryck	SMART	Ende.netbook	Göra reklam för deltagarna.
3:55-4:00	ExperimentYta	Utvärdering	Få tillbaka en utvärdering	Frågeformulär	Utvärderings-formulär	Påpeka att även negativ är bra kritik.

## Innan laborationen

- Låna nyckeln till hallen (nyckeln med Toa-chip)
- 1x USB-sticka (med alla presentationer)
- Ev. Låna kamera eller videokamera
- Se till att ha deltagarlista tillhands
- Skriva ut handouts (Java)
- Förbereda hallen
  - Se till att hallen är städad
  - Ev. Starta värmen
  - Lägga sittsäckarna ordentligt
  - Ställa ut soppsåsar
  - Förbereda Demo-Yta
    - Ställa ut bord
    - Kontrollera batterier
    - Ställa ut robotarna (ev. Med hjälp av kursmappen)
    - Förbereda fjärrkontroll
    - Ordna beskrivningsblad
    - Förbereda Experiment Yta
    - Ställa i ordning borden
    - Ställa ut tillräckligt med stolar (om möjligt inte vid projektorn)
    - Fördela ut förgreningsdosor.
    - Datorer och NXT-lådor fördela (**Varje grupp: 1x Dator, 1x vit låda och 1x blå låda**)
    - USB-Kabel fördela
    - Testa SMART-board med presentations-datorn, se till att alla nödvändiga presentationer är förbereda och hemsidan är gömd.
  - Förbereda „Klassrummet“
    - Ställa i ordning bord
    - Ställa ut tillräckligt med stolar
    - Testa SMART-board med presentations-datorn, se till att alla nödvändiga presentationer är förbereda och ev. att hemsidan är gömd.
  - Förbereda duell-Arena
    - Dekorera inför rätt laboration
    - Förbereda „kärnkraftverket“
  - Förbereda NXT-lådorna
    - Kolla så att alla bitar finns i lådorna.
    - Kontrollera så att NXTna är laddade
    - Tömminnen på gamla program
    - Ev. Ladda ner kalibreringsprogram på NXTna
    - Kolla så att LEJOS är installerat på alla NXT
  - Förbereda datorerna
    - Alla datorer ska ha en mus och en elkabel
    - Är Java och LeJoS-mjukvarorna installerade?
    - Se till att alla konstruktionsbeskrivningar är på alla datorer
    - Internet-inställningar?
- Förbereda NXT-enheten

- På alla enheter ska LEJOS installeras(använd nxj flash som ska ligga på skrivbordet på alla datorer). Installera genom att förbinda enheten med datorn, öppna nxj flash filen och skriv helt enkelt bara 1 när programmet frågar efter något.
- Utvärdering
  - Skriva ut utvärderingen



# Detaljerat program

## 1. Hälsning

- Välkomna alla elever och berätta lite kort om dagen.

## 2. Demo-Yta

- Eleverna har möjlighet att själva testa olika robotar

## 3. Organisera grupper

- Organisera grupper om 2 eller 3 elever

## 4. Introduktion i robotik

- Hålla presentation med hjälp av dokumentet *Einführung Robotik*
- Detaljerad beskrivning av presentationen:
  - **Sida 1: Robotik.** Vad är robotik? Har någon ett förslag? (här kan vi skriva på presentationsunderlaget under pågående presentation). Lämna lite tid och vänta på svar från eleverna.
  - **Sida 2: Vad är egentligen en robot?** Under texten är slaw. : robota=arbete gömt. Hur skulle du beskriva en robot? De mesta av oss har sett robotar tidigare i olika situationer. Vi använder begreppet ganska brett utan att de flesta av oss egentligen inte vet vad som menas med det, vad det betyder och var det kommer ifrån. Har någon av er die vad en robot är? (ge associationer) Slovakiskt ord: „arbeta“ (i sinne för kroppsligt arbete). I dagens betydelse användes begreppet redan 1920 i ett tjeckiskt teaterstycke som handlar om människoliknande maskiner so mär skapade för att underlätta för människan.
  - **Sida 3: Robotens beståndsdelar och dess motsvarigheter**  
Vad är en robot? En maskin som kan ta över människans arbete. Vad innebär detta exakt? Det finns så många exempel av robot arter, men alla har dock gemensamt att: en robot är något som sätts i interaktion med den fysiska världen med bas av sensorer, motorer och informationsöverföring. Jämför med människor (sensorer/sinnen, motorer/muskler och hjärna/NXT-enhet)
  - **Sida 4: Interaktion.** Men vad är en interaktion? Kan vara alla sorters växelverkande. Även hos oss människor: tala, höra, känna, gripa, gå... mekaniskt: ex känna temperatur. Kommutativt: skicka/ta emot signaler osv.
  - **Sida 5: Robotar är speciella maskiner.** Robotar är stationära eller mobila maskiner, som kan efter bestämda program genomföra fastställda program men till skillnad från maskiner kan man ändra programmet.
  - **Sida 6: Robotregler**
    0. En robot får inte skada mänskligheten, eller genom passivitet tillåta att människan kommer till skada.
    1. En robot får inte skada mänskligheten eller genom passivitet tillåta att människan kommer till skada, förutom att han bryter mot den nollte lagen.
    2. En robot måste lyda order från folket – om inte sådana order strider mot den nollte eller första lagen.
    3. En robot måste skydda sin egen existens, om inte denna strider mot den nollte, första eller andra lagen.

Isaac Asimov med hjälp av John W Campbell skapade dessa regler 1940 då var dessa regler endast tre men 1986 lades den nollte regeln till.

- **Sida 7: Robotarter.**

I varje hörn finns en art gömd. Överst till vänster: humanoid, under till vänster: leksak. Under till höger: industri. Under till höger: prospektering (militär).

Vilka typer av robotar finns det? Det finns inga förslag på presentationen men kan eleverna inte komma på några arter så finns det: humanoid (robotarna är möjligen liknande människan och mycket komplicerade (som är förverkligade de senaste åren)). Industriella (t.ex. används av nästan alla biltillverkare) dessa tar över arbete, de är snabbare och/eller noggrannare än människan. T.ex. svetsning, hålla, nitning och lödning. Servicetobotar ((dit hör även leksakerna) Servicerobotar är robotar som kan utföra hushållsarbeten som t.ex. dammsuga eller klippa gräsmattan En ganska speciell robot är „Care-O-bot“. Denna robot är en tjänsterobot, som hjälper sjuka, gamla och funktionshindrade i deras liv. Till serviceroboten tillhör även leksaksrobotar som t.ex. „Aibo“ från Sony, ett konstgjort husdjur som springer, sitter och kan leka med en boll. Projekteringsrobotar (militär) används av t.ex. polisen och militären. Dessa har oftast en uppräpningsfunktion, eller tar över uppgifter som skulle vara skadliga för människan (t.ex. Arbeta med bomber). En välkänd är NASAs Mars-Rover , „Big Dog“, en mekanisk arbetsträl för militär situationer. Robotar kan användas i många olika situationer.

Alla robotar kommer inte behöva nämnas i laborationen, men att känna till dem är bra för att vara förbered för eventuella frågor.

- **Sida8: Lego Mindstorms.**

Och nu vill vi använda byggsatserna, dvs. LEGO Mindstorms systemet. I detta system hittar vi allt vi behöver för att bygga och programmera en robot. Vi behöver lego delar , sensorer och en programmerbar enhet. Det fanns ett flertal olika sensorer, t.ex. tryck, ljud och ljus.

## 5. Introduktion till räddningsinsatsen

- Genomföra presentationen av uppgiften med att hjälpa av dokumentet *Rettungseinsatz*
- Detaljerad beskrivning av uppgiften
  - **Sida 1: Räddningsinsats**

Välkomna till RoboScope kursen Räddningsinsats. Det finns flera olika uppgifter för en räddningsrobot.
  - **Sida 2: Foton från Fukushima**

T.ex. kan en robot arbeta i ett kärnkraftverk. Efter en olycka i början av 2011 arbetade robotar i kärnkraftverket i Fukushima.
  - **Sida 3: Foton av robotar som arbetade inuti kärnkraftverket**
  - **Sida 4: Änne ett par foton av robotarna i arbete**
  - **Sida 5: Våran robot ska kunna göra följande:** hitta strålningskällan (infrarödstrålning), navigera, mäta strålningen och köra till strålningskällan. Vi kommer att arbeta i grupper om 2 eller 3 elever. Varje robot kommer att bygga sin egen robot i Lego Mindstorms och sedan programmera denna i Java. **Observera att uppgifterna inte behöver genomföras i denna ordning!**
  - **Sida 6: Vi behöver:** två sensorer (ultraljud och infraröd), motorer (så att roboten kan röra sig) och även den programmerbara enheten.
  - **Sida 7: Presentera de olika valmöjligheterna.**

Vad måste vi tänka på när vi bygger en robot? Stabilitet, konstruktion och estetik. Diskutera tillsammans olika aspekter. Beskriv de olika modulerna och förklara att de ska välja själv

vilken robot de vill bygga, och att de ska motivera sina val. Varför gör ni så? Fördelar? Nackdelar? De ska i slutet av laborationen visa sin robot och då kunna förklara hur de tänkt osv.

- **Sida 8: Nu börjar det!**

## 6. Diskussion och byggfas

- Eleverna börjar med en diskussion om vilken modul de vill bygga. Eventuellt behöver eleverna hjälp för att komma igång med diskussionen. De har tillgång till papper och penna om de vill anteckna något.
- När de valt vilken robot de vill bygga är det bara att börja. Det finns bygginstruktioner i PowerPoint till alla olika moduler: *Konstruktion1* till *Konstruktion3*.

## 7. Introduktion till Java och LeJOS (kanske möjligt att göra den även med NXT-G)

- Genomföra presentationen med hjälp av dokumentet *Einführung Java*
- Detaljerad beskrivning av presentationen

- **Sida 1: Introduktion till Java**

Idag ska vi programmera i Java. Jag tror att alla av er redan har programmerat med Java tidigare. Men det finns några nya saker vi behöver kunna för att kunna programmera NXT-enheten.

Vad är egentligen programmering? Vad tänker ni? (Möjligt att lägga till information på presentationsunderlaget).

Alla har minst en åsikt om vad programmering är, idag ska vi gå igenom vad programmering är. Viktiga slagord är: order, upprepning/order, självständigt/automatiskt, tänka/intelligens.

- **Sida 2:**

Datorer är dumma

*Eftersom: de endast kan förstå enkla steg.*

Datorer är lata

*Eftersom: de inte gör något som man inte sagt åt dem att göra.*

Datorer behöver enkla besked efter varandra. Programmering innehåller instruktioner eller förklaringar som maskinen/roboten ska följa. Man kan programmera i flera olika språk, idag ska vi arbeta med det objektorienterade språket Java. Det betyder att man måste programmera i enkla steg och göra programmet tydligt och organiserat. Vi kan säga till en person att göra något som den automatiskt kommer förstå utan djupare beskrivning, men det klara roboten inte. Vi måste ta allt i små steg.

Java är ett objektorienterat programmeringsspråk (en sats objekt som interagerar med varandra). När vi vill använda motorer och sensorer använder vi LeJOS-paketet som kan kommunicera med NXT. (LeJOS = LEGO Java Operation System). Vi måste importera lejos: `Import lejos.nxt.*`

- **Nästa steg skulle då vara att öppna Eclipse.** Visa hur man gör ett program och hur man kan använder de olika funktionerna. Gör detta interaktivt tillsammans med eleverna. Här finns ett exempel (men då detta steg görs i huvudsak av eleverna är det möjligt att det ser annorlunda ut).

```
import lejos.nxt.*;
import lejos.nxt.addon.IRSeekerV2;
import lejos.nxt.addon.IRSeekerV2.Mode;
public class Test {
    public static void main(String [] args) throws Exception{
        while (!Button.ESCAPE.isPressed()){
```

```

IRSeekerV2 Sucher = new IRSeekerV2(SensorPort.S2, Mode.AC);
//Ger infrarödsensorn namnet Sucher.
UltrasonicSensor sonic = new UltrasonicSensor(SensorPort.S2);
Sucher.setAddress(0x10);
//Nödvändiga inställningar för sensorn.
    int direction = Sucher.getDirection();
    int distanz = sonic.getDistance();
        if (distanz > 20){
            Motor.A.forward();
            Motor.B.forward();
            Thread.sleep(1000);
        }
        else{
            LCD.drawString(direction,0,1);
        }
    }
}

```

**Slingor:** En slinga upprepar en anvisning så länge kravet är uppfyllt eller tills en order om avbrott kommer. For-slinga: talvariablerna anges inom for-slingan och efter det följer ordern, därefter kommer den så kallade indexeringsuttryck som ger talvariabeln ett nytt värde. For (int x = 8; x >= 0; x-). If-else anvisning: if används för att realisera ett krav. Bara när kravet inträffar kommer programmet följa anvisningarna efter if, annars kommer det följa anvisningarna efter else.

**leJOS.** Det finns fler olika klasser i LeJOS-paketet, men idag behöver bara ett par av dessa. Det finns t.ex. följande klasser: ljussensorn, ultraljudsensorn, motor, LCD och trycksensor.

**Display.** På displayen kan man skriva både textmeddelanden och mätvärden. Det är en LCD-display med 100\*64 pixlar. Ett tecken har höjden 8 px och bredden 6 pr, så på varje rad kan man skriva 16 tecken. Normalt kan man alltså skriva 8 rader, med start på rad 0. Början av koordinatsystemet är i det övre vänstra hörnet.

#### Klassen LCD:

DrawString(string text, int x, int y)

Clear()

Scroll()

**Sensorer.** NXTs sensorer är deras ögon och öron och kan samla information till roboten från omvärlden. Standardpaketet innehåller fyra sensorer: en ljussensor, en ultraljudssensor, en ljudsensor och en beröringssensor. Beröringssensorn kan bara säga om den är tryckt eller inte, medan de andra ger ett värde tillbaka.

**Ultraljudssensor.** Ultraljudssensorn mäter avståndet till ett objekt, i cm. Den kan mäta inom en vinkel på ca 30° (-15° till + 15°) kantiga föremål med en glansig yta. Det maximala avståndet är ca 250 cm, när den inte hittar något föremål framför sig blir det stora mätfel. Vid ett avstånd på 150 cm ska föremål befinna sig inom en vinkel av 20° från sensorn. Minskar avståndet mellan sensorn och föremålet kan den även upptäcka mer roterade objekt.

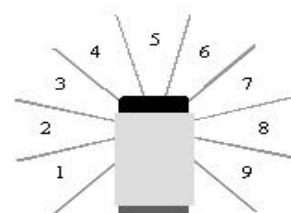
#### Klassen UltrasonicSensor:

int getDistance()

**Den infrarödsensorn.** Den infrarödsensorn mäter strålningen från en infraröd källa. IRSeeker använder en digital signal och filtrera bort bakgrundssignaler. Med IRSeeker kan vi få ett värde (se bilden bredvid) eller grader (mellan -180 och 180) tillbaka.

#### Klassen IRSeekerV2:

getAngle()



getDirection  
getSensorValue()

**NXT-motor.** Motorerna till NXT har en maximal vinkelhastighet på 900° per sekund, dvs. ca 2,5 varv per sekund. Motorerna kan roteras i steg per varje grad. Hastigheten bestäms utifrån användbar kraft. Då motorerna är servomotorer har de interna rotationssensorer som mäter hur många grader de kan rotera. På detta sätt kan den passerade sträckan beräknas. Därför kan även motorerna användas som sensorer.

Klassen motor:

setSpeed(int gps)  
setPower(int kraft)  
forward()  
backward()  
stop()  
rotate(int vinkel)  
int getTachoCount()  
shutdown()

- Öppna dokumentet **Aufgabe**. Vår robot ska kunna: röra sig, köra framåt, köra bakåt, vända x ° grader, hitta strålningskällan, röra sig till målet och undvika hinder. Diskutera uppgiften, vad varje sensor ska göra och hur motorerna ska användas. Här kan det vara ett ide att ge några tips som man kan ha sätt vara bra för eleverna att känna till för att underlätta arbetet.

## 8. Programmeringsfasen

- Eleverna ska programmera sina robotar på egenhand. Eleverna har hjälp av en handout om Java. Assistenterna cirkulerar för att hjälpa eleverna.
- Försök att inte ge några svar utan öppna för nya tankar hos eleverna.
- Dela ut Handouts

## 9. Duell-Arena

- Alla grupper ska visa sina robotar och förklara varför de valt att göra så. Här är det möjligt att det finns ett behov att ställa frågor för att få i gång diskussionen.

## 10. Plocka isär robotarna. Visa hemsidan samtidigt.

- Vara tydlig med att sorteringen i lådorna ska hållas så att det är lätt att bygga en robot nästa gång också.

## 11. Slut

- Använd dokumentet **Ende**
- Diskussioner, frågor och framtida studier.
  - **Sida 1: Slutligen.**
  - **Sida 2: Diskussion och bilders:** Vad har ni lärt er? Vad skulle ni göra annorlunda annars? Varför? Använd möjligheten med att skriva på SMART-boarden. (Här finns det ingen direkt mall utnyttja situationen).
  - **Sida 3: Yrke: att bygga robotar.** Frågor till eleverna: vad behöver man kunna för att bygga en robot? Vilka ämnen ska man ha studerat? Vilka har som uppgift att bygga en robot?
  - **Sida 4: I robotar rymms...** Datavetenskap... Matematik... Fysik... Kemi ... Biologi/Bioteknik... Elektroteknik... Maskin- och trafikteknik... Mekatronik... Energiteknik... Materialvetenskap... Teknikkonstruktion/Design

- **Sida 5: Aktuella Projekten** t.ex. medicinska robotar, Care-O-bot, elektroniska vägar för lastbilar, rörelserobotar som kan arbeta i ett samhälle med människor. (Här är det ganska lätt att lägga till eller ändra till andra projekt utifrån vilka projekt som intresserar er). Här finns det ännu en möjlighet att intressera eleverna.
- **Sida 6: Frågor?**
- **Sida 7: Besvara utvärderingen**

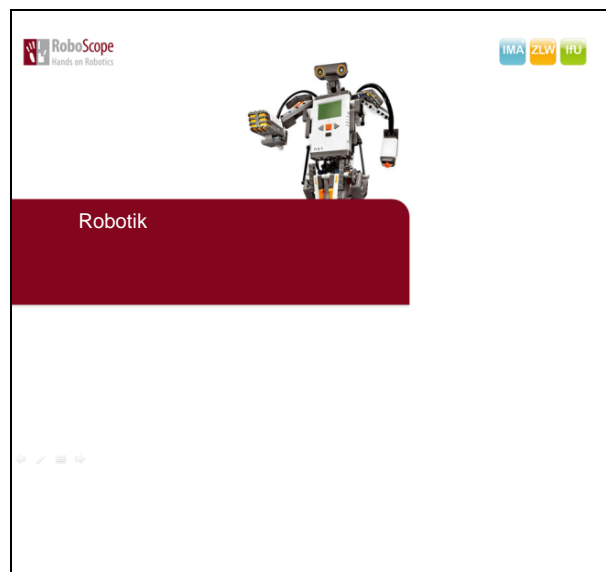
## **Tips!**

- Se till att alla datorer är stängda under programmeringsföredraget

## **Efter laborationen**

- Städa hallen
  - Skräp, lådor och datorer
  - Stänga av alla demo-robotar
- Lämna tillbaka kameran
- Lämna tillbaka USB-stick
- Läs igenom och sammanfatta utvärderingen
- Lämna tillbaka nyckeln

## A.2 Presentations for the laboratory work





RoboScope Hands on Robotics Bestandteile eines Roboters und ihre Gegenstücke

· Aktoren



Muskeln



· Sensoren



Sinne



· Informationsverarbeitung



Gehirn



RoboScope Hands on Robotics

Interaktion

- Sprechen, Hören, Funk, etc (kommunikativ)
- Greifen, Treten, Bohren, etc (mechanisch)
- Temperatur fühlen, Strahlung messen, etc (physikalisch)



RoboScope Hands on Robotics

Roboter sind besondere Maschinen:

Roboter sind stationäre oder mobile Maschinen, die nach einem bestimmten Programm festgelegte Aufgaben erfüllen, aber im Vergleich mit einer Maschine kann man das Programm verändern.



## Robotergesetze

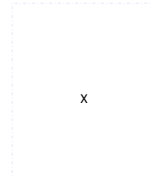
„0. Ein Roboter darf die Menschheit nicht verletzen oder durch Passivität zulassen, dass die Menschheit zu Schaden kommt.

1. Ein Roboter darf keinen Menschen verletzen oder durch Untätigkeit zu Schaden kommen lassen, außer er verstieße damit gegen das nullte Gesetz.

2. Ein Roboter muss den Befehlen der Menschen gehorchen – es sei denn, solche Befehle stehen im Widerspruch zum nullten oder ersten Gesetz.

3. Ein Roboter muss seine eigene Existenz schützen, so lange dieses sein Handeln nicht dem nullten, ersten oder zweiten Gesetz widerspricht.“

*I. Asimov*



## Robotarten



## Mindstorms NXT





## Das Programm Heute

- Begrüßung
- Demostraße
- Einführung Robotik und das Aufgabe
- Bauphase
- Einführung Java
- Programmierungsphase
- Wettkampf
- Ende

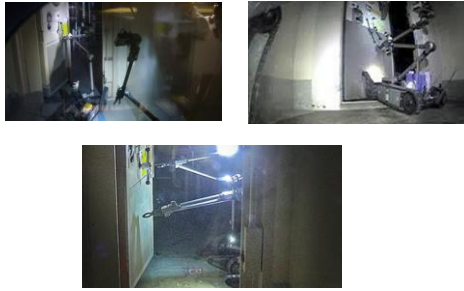


# Rettungseinsatz



Beispiel





### Unser Roboter soll:

- Navigieren
- Hindernisse abweichen
- Die Strahlung messen
- Das Leck lokalisieren



### Dazu brauchen wir

- Sensoren
- Aktoren
- Der NXT-Block



## Wahlmöglichkeiten



## Jetzt geht s los

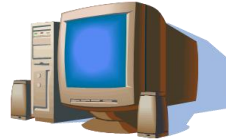




## Programmieren in Java



- Computer sind dumm  
*Weil: sie nur ganz kleine Schritte verstehen können.*
- Computer sind faul  
*Weil: sie nichts tun, wenn man es ihnen nicht vorher sagt.*





## Unser Roboter soll können:

- Bewegung
  - Vorwärts/Rückwärts fahren
  - Drehen um  $x$
- Die Strahlenquelle finden
- Bewegung zum Ziel
- Hindernisse abweichen



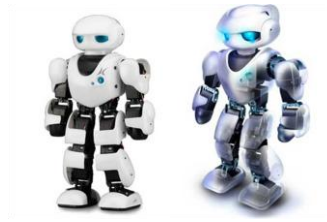
Zum Schluss...



## Diskussion



## Beruf: Roboter bauen



## In Robotern steckt...

- ... Informatik
- ... Mathematik
- ... Physik
- ... Chemie
- ... Biologie/Biotechnologie
- ... Elektrotechnik
- ... Maschinen- und Verkehrstechnik
- ... Mechatronik
- ... Energietechnik
- ... Werkstoffwissenschaften
- ... Technikgestaltung/Design



## Aktuelle Projekte

- Elektronische "Schienen" für LKW
- Medizinische Roboter
- Care-O-bot
- Bewegungsroboter die sich in der Nähe von Menschen bewegen



## Fragen?



# Feedback



## A.3 Building instructions



RoboScope  
Hands on Robotics

IMA ZLW IfU

4

RoboScope  
Hands on Robotics

IMA ZLW IfU

5

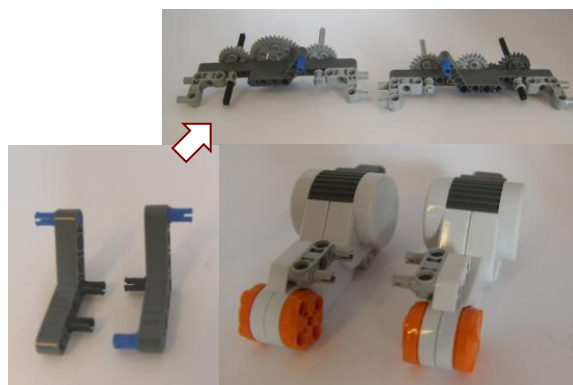
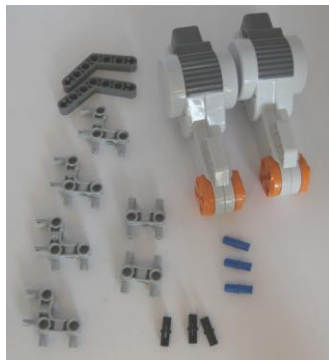
RoboScope  
Hands on Robotics

IMA ZLW IfU

6

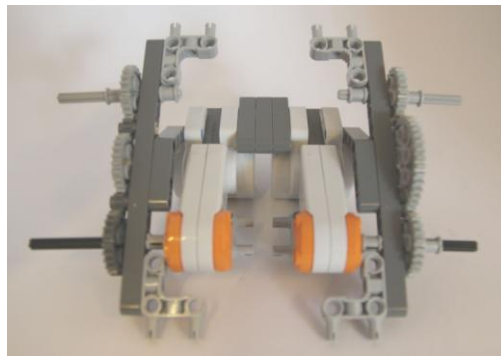
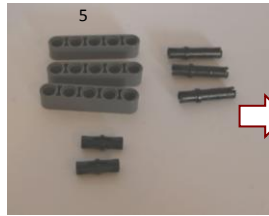


neue Teilen hinzufügen:

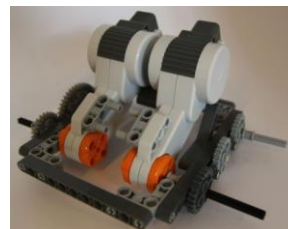
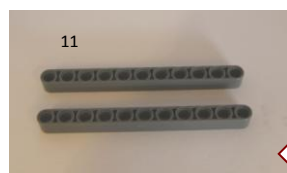




neue Teilen hinzufügen:



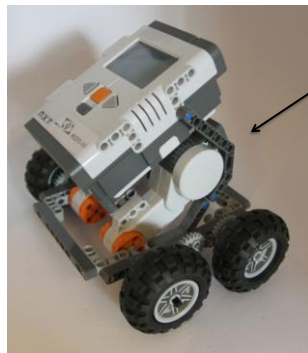
neue Teilen hinzufügen:



neue Teilen hinzufügen:

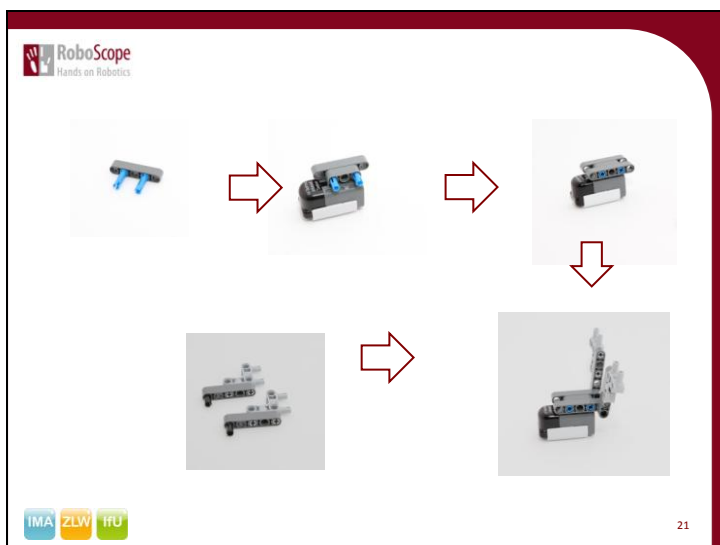


neue Teilen hinzufügen:



neue Teilen hinzufügen:







#### benötigte Kabel:

- 1 kurzes Kabel (ca. 15 cm)
- 3 mittlere Kabel (ca. 35 cm)

#### Verbindungen

- Infrarotsensor mit Anschluss 1 (kurzes Kabel)
- Ultraschallsensor mit Anschluss 2
- Rechter Motor mit Anschluss A
- Linken Motor mit Anschluss B

→ Kabel möglichst zu einer Spirale rollen!  
(oder ähnlich geschickt befestigen)

**VIEL SPAß!**



# Konstruktion

## Hinweis



**Bedeutet:** Die Stange ist  
6 Löcher oder Noppen lang

*... und los geht's!*

## benötigte Teile:

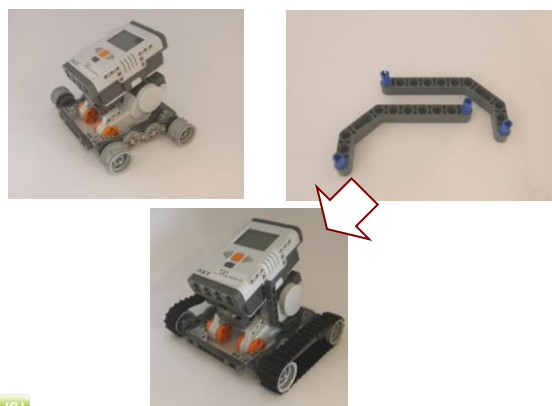
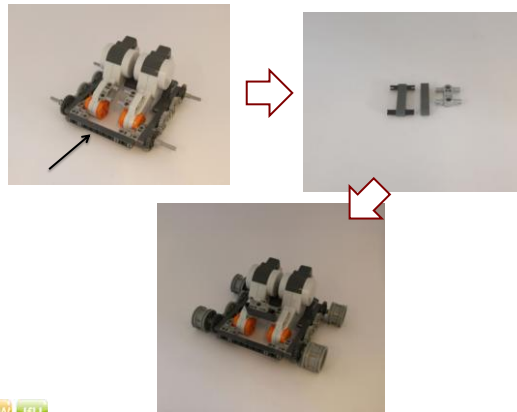




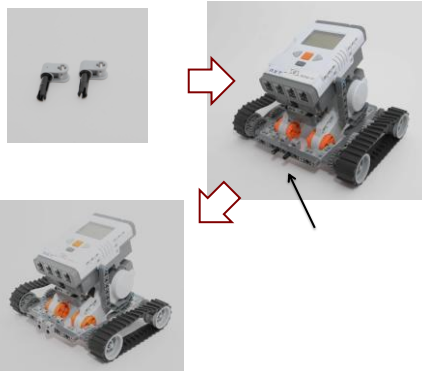




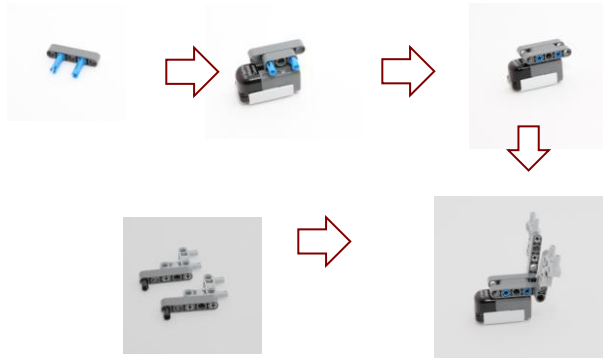
neue Teilen hinzufügen:



neue Teile hinzufügen:



neue Teile hinzufügen:



### Benötigte Kabel:

- 1 kurzes Kabel (ca. 15 cm)
- 3 mittlere Kabel (ca. 35 cm)

### Verbindungen

- Infrarotsensor mit Anschluss 1 (kurzes Kabel)
- Ultraschallsensor mit Anschluss 2
- Rechter Motor mit Anschluss A
- Linken Motor mit Anschluss B

→ Kabel möglichst zu einer Spirale rollen!  
(oder ähnlich geschickt befestigen)

**VIEL SPAß!**



# Konstruktion

## Hinweis



**Bedeutet:** Die Stange ist  
6 Löcher oder Noppen lang

*... und los geht's!*

## benötigte Teilen:

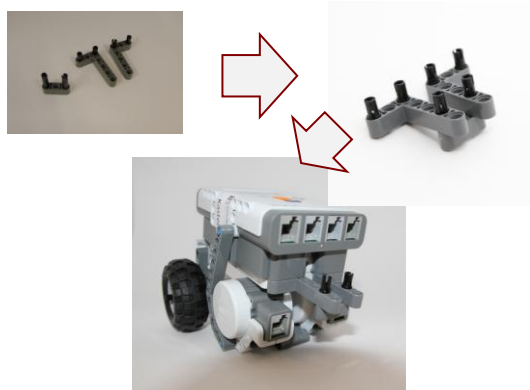
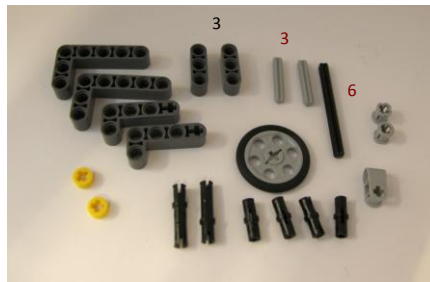








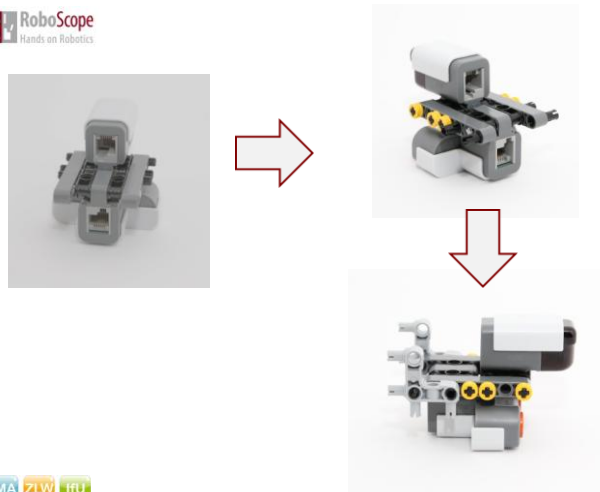
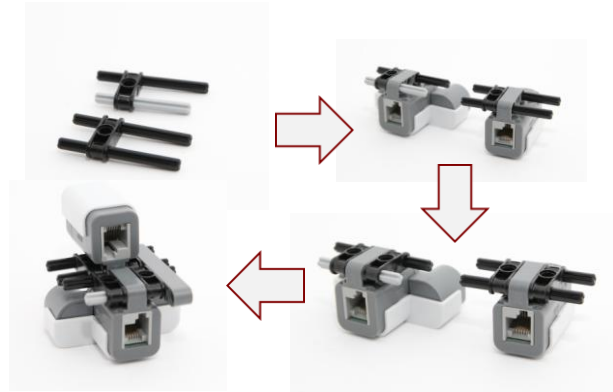
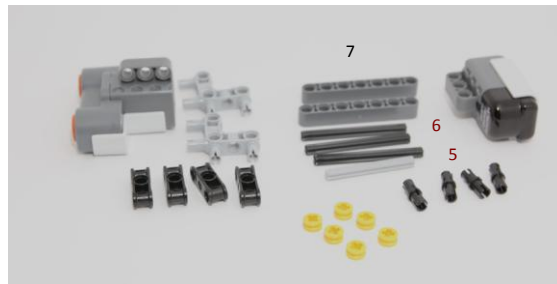
neue Teile hinzufügen:







neue Teile hinzufügen:





#### Benötigte Kabel:

- 1 kurzes Kabel (ca. 15 cm)
- 3 mittlere Kabel (ca. 35 cm)

#### Verbindungen

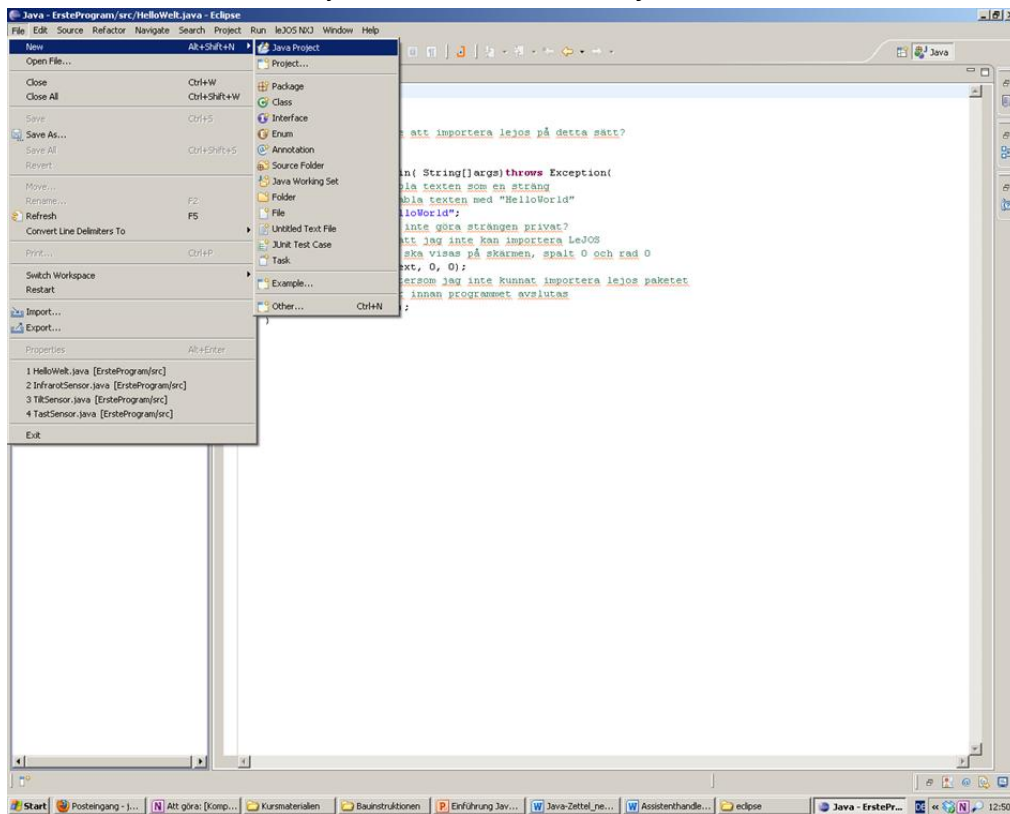
- Infrarotsensor mit Anschluss 1 (kurzes Kabel)
- Ultraschallsensor mit Anschluss 2
- Rechter Motor mit Anschluss A
- Linken Motor mit Anschluss B

→ Kabel möglichst zu einer Spirale rollen!  
(oder ähnlich geschickt befestigen)

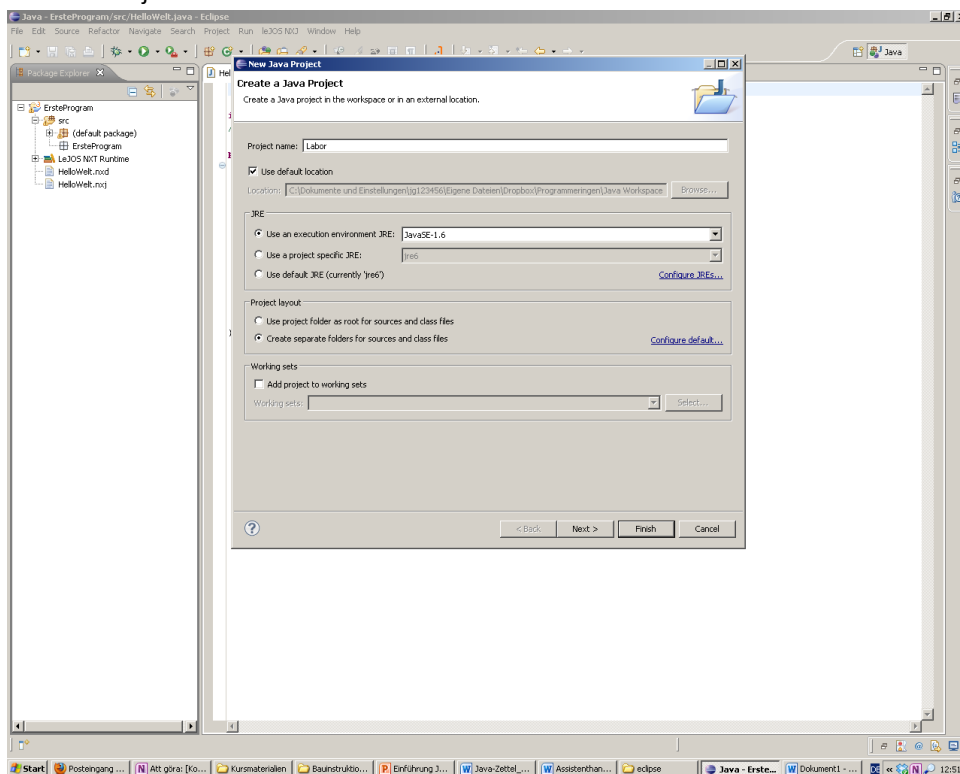
**VIEL SPAß!**

## A.4 Handout Programmierung / Stencil programmering

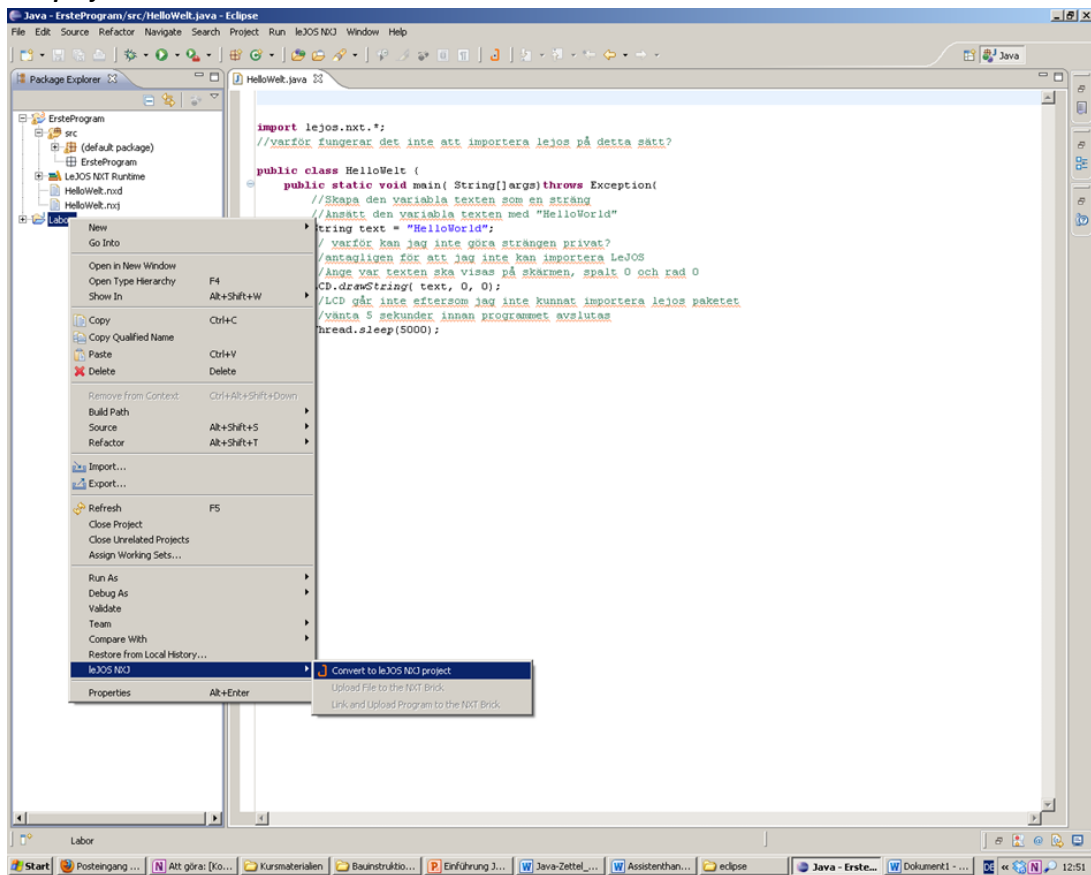
1. Anfang mit **Eclipse** öffnen (liegt auf dem Schreibtisch).
2. Dann machen wir ein neu Project. **File** → **New** → **Java Project**



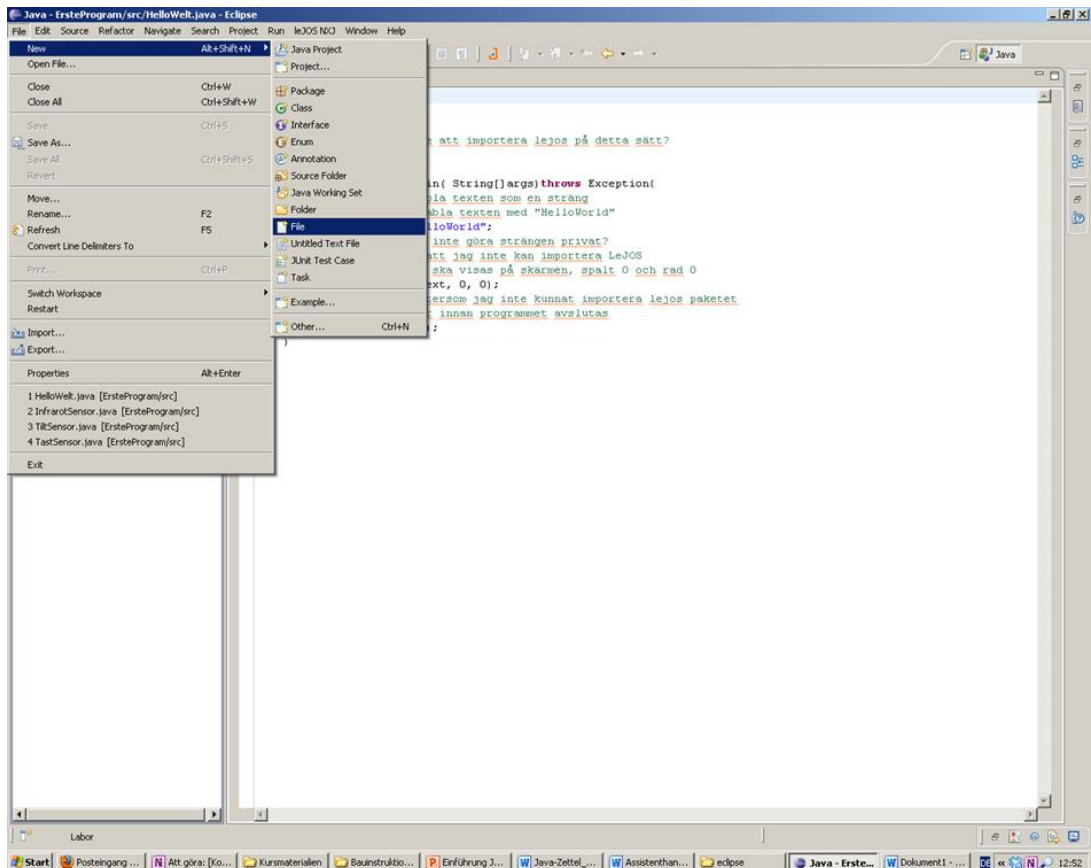
3. Geb den Projekt ein Name und dann **Finish**.



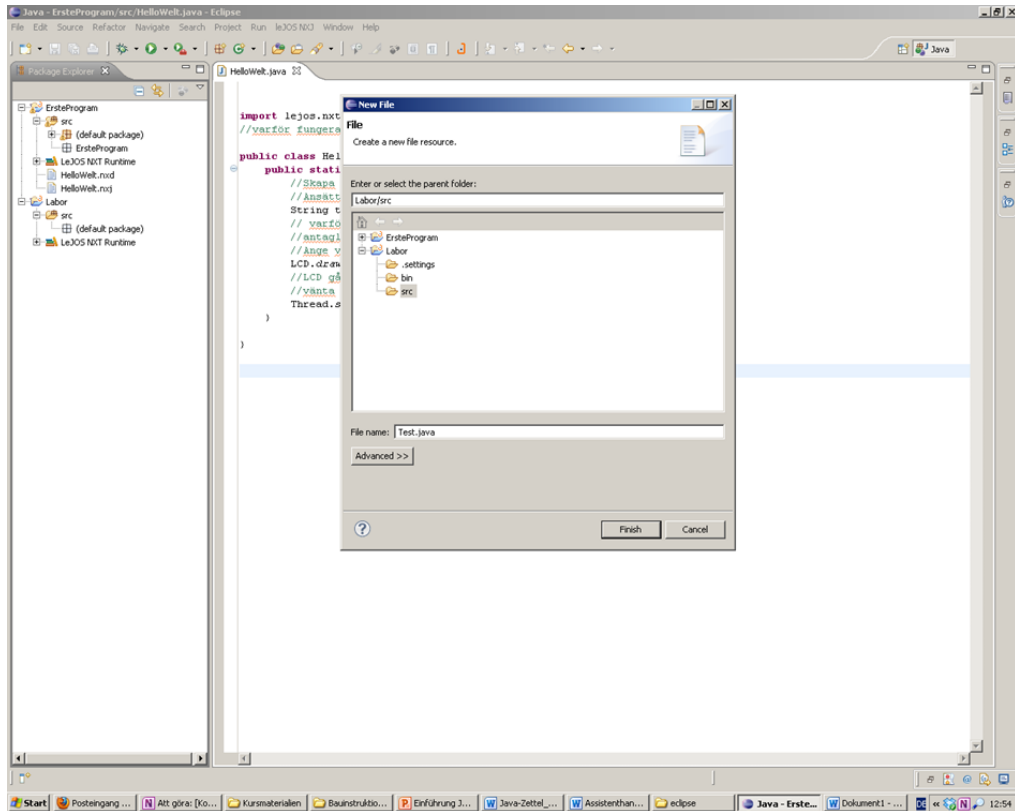
4. Konvert das Projekt zum Lejos Project. Drücken auf dem **Maus rechtsklick** → **leJOS NXJ** → **Convert to NXJ project**.



5. Jetzt würden wir ein neu File machen. **New** → **File**



6. Liegt den file an dein project und *src*. Geb dein file ein name *xxx.java*. Und dann **Finish**.



## Struktur ein Programm machen:

```
import lejos.nxt.*;
// Importiert die wichtigsten Lejos Bibliotheken.

public class xxx{
    public static void main( String[]args) throws Exception{
        ...
        ...
    }
}
```

### Schleifen:

```
while (Bedingung) {...}
```

```
for( int i = 0; i < 10; i++) {...}
```

```
for( Deklaration; Bedingung; Veränderung) {...}
```

### Abfragen:

```
if (Bedingung) {...}
```

```
else{...}
```

## Verschiedene Funktionen:

### Das Display

Auf dem Display können sowohl textnachrichten als auch Messwerte oder Sensorwerte ausgegeben werden. Auf jede Zeile können wir 16 (0 bis 15) Zeichen schreiben und das gibt 8 Zeile (0 bis 7). Der Ursprung des Koordinatensystems befindet sich, wie in Java üblich, am oberen linken Displayrand. Die für das Display verwendete Klasse ist LCD.



### Die Klasse LCD

Methode	Beschreibung
<code>drawString(String text, int x, int y)</code>	Gibt einen Text auf den Bildschirm aus. Der Parameter ist für den Text, x ist die Spalte und y die Zeile.
<code>drawString(int zahl, int x, int y)</code>	Gibt eine Ganzzahl auf dem Schirm aus. Parameter Zahl ist für die Zahl gedacht, x definiert die Spalte, y

Clear ()	die Zeile. Diese Funktion löscht das Display. Sie benötigt keine Parameter.
----------	--

## Der NXT-Motor

Die Motoren des NXT haben eine maximale Winkelgeschwindigkeit von 900° pro Sekunde, was in etwa 2,5 Umdrehungen pro Sekunde entspricht. Die Motoren des NXTs können in 1°-Schritten angesteuert werden. Da die Motoren Servomotoren sind, verfügen sie über interne Rotationssensoren, die messen, um wie viele Grad sie gedreht wurden. Auf diese Weise kann zum Beispiel ein zurückgelegter Weg berechnet werden. Somit können die Motoren sowohl als Aktoren als auch als Sensoren genutzt werden.



## Die Klasse Motor

Methode	Beschreibung
setSpeed(int gps)	Setzt die Motorgeschwindigkeit in Grad/Sekunde (gps) wobei 900 dem Maximum entspricht
setPower(int kraft)	Setzt die Motorkraft, der Wertebereich der Variablen kraft liegt zwischen 0 und 100.
Forward()	Lässt den Motor endlos vorwärts fahren.
Backward()	Lässt den Motor endlos rückwärtsfahren.
Stop()	Veranlasst einen sofortigen Stopp und hält den Motor auf Position (lock). Beendet z.B. <i>forward()</i> Funktionen.
rotate(int winkel)	Rotiert den Motor solange, bis der Winkel <i>winkel</i> erreicht ist, dann schaltet der Motor ab.

## Der Ultraschallsensor

Der Ultraschallsensor misst den Abstand zu Objekten in cm. In einem Winkel von ca. 30° (-15° bis + 15°) werden vornehmlich eckige Gegenstände mit einer glatten Oberfläche erkannt. Die maximale Entfernung dabei entspricht ca. 250 cm- Sobald die Ultraschallwellen nicht frontal auf den Gegenstand treffen, ergeben sich sehr schnell große Messfehler. Bei einer Entfernung von 150 cm sollte das zu erkennende Objekt nicht weiter als 20° vom Winkel des Ultraschallsensors gedreht sein. Verringert sich die Entfernung zwischen Gegenstand und Sensor, werden auch stärker gedrehte Objekte erkannt.





```
UltrasonicSensor ii = new UltrasonicSensor(SensorPort.S1);
```

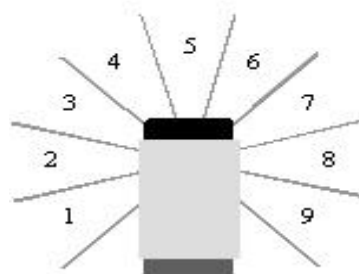
Erstell den Ultraschallsensor mit ii.

### Die Klasse UltrasonicSensor

Methode	Beschreibung
Int getDistance()	Gibt die Entfernung zu einem gefundenen Objekt zurück.

### Der Infrarotsensor

Der Infrarotsensor misst die Strahlung von einem Infrarot Quelle. Das IRSeeker benutzt ein digital Signal und filtern die empfängt Signalen. Mit der IRSeeker können wir ein Wert (siehe das Bild unten) oder Graden (zwischen -180 und 180) als Rückwerte bekommen.

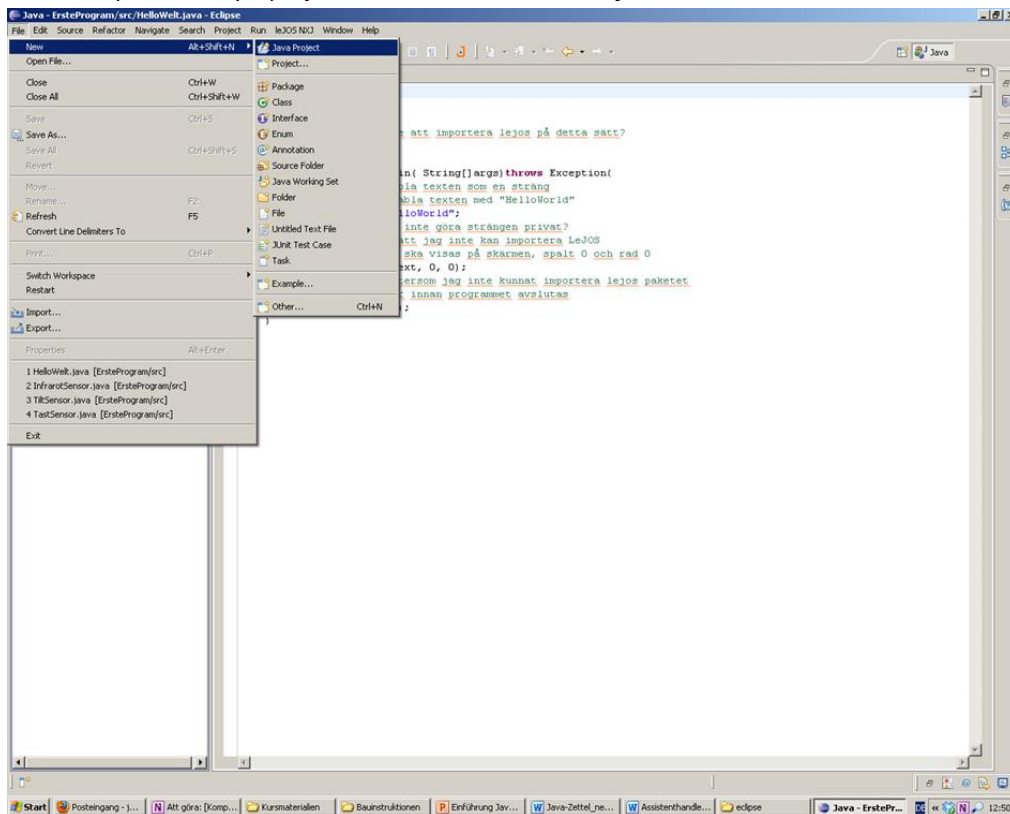


```
import lejos.nxt.addon.IRSeekerV2;
import lejos.nxt.addon.IRSeekerV2.Mode;
Importiert die Bibliotheken für den Infrarotsensor.
IRSeekerV2 iii = new IRSeekerV2(SensorPort.S2, Mode.AC);
Erstellt den Infrarotsensor mit dem Namen iii.
iii.setAddress(0x10);
Notwendige Einstellung für den Sensor.
```

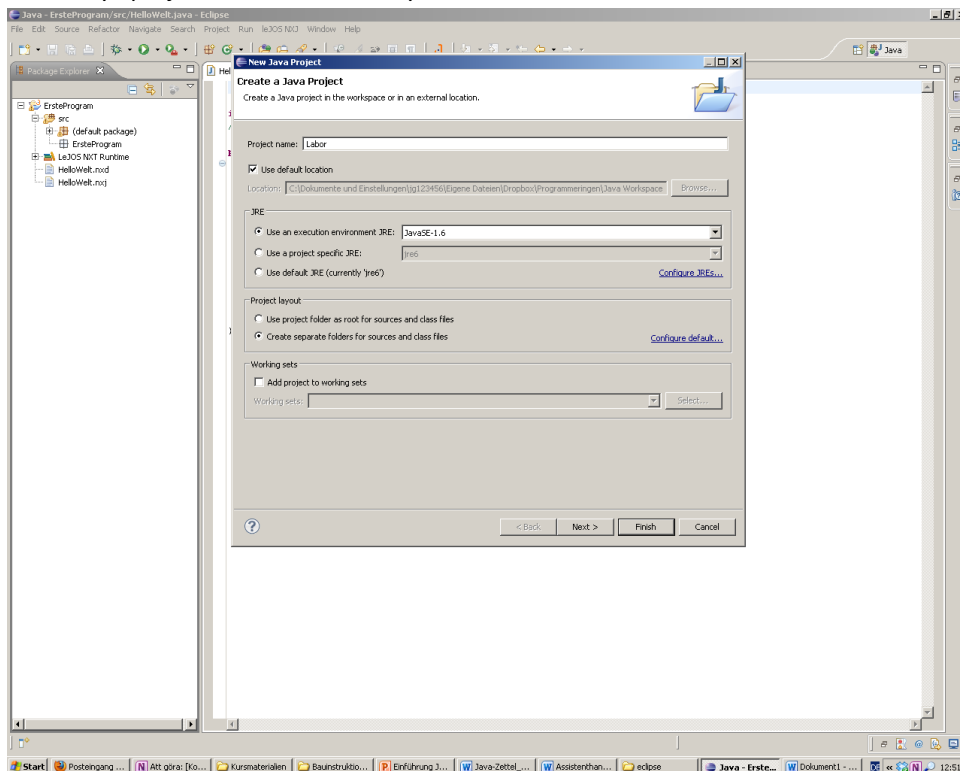
### Die Klasse IRSeekerV2

Methode	Beschreibung
Float getAngle(boolean blocking)	Gibt die Winkel zum Ziel(zwischen -180 und 180 Graden) zurück.
Int getDirection()	Gibt die Richtung mit dem höchsten infraroten Wert an ( zwischen 1 und 9)
Int getSensorValue(int id)	Gibt die Intensität des Sensors i an. Die Sensoren sind nummeriert von 1-5 und liegen an den Stellen, 1,3,5,7 und 9 auf dem Bild.

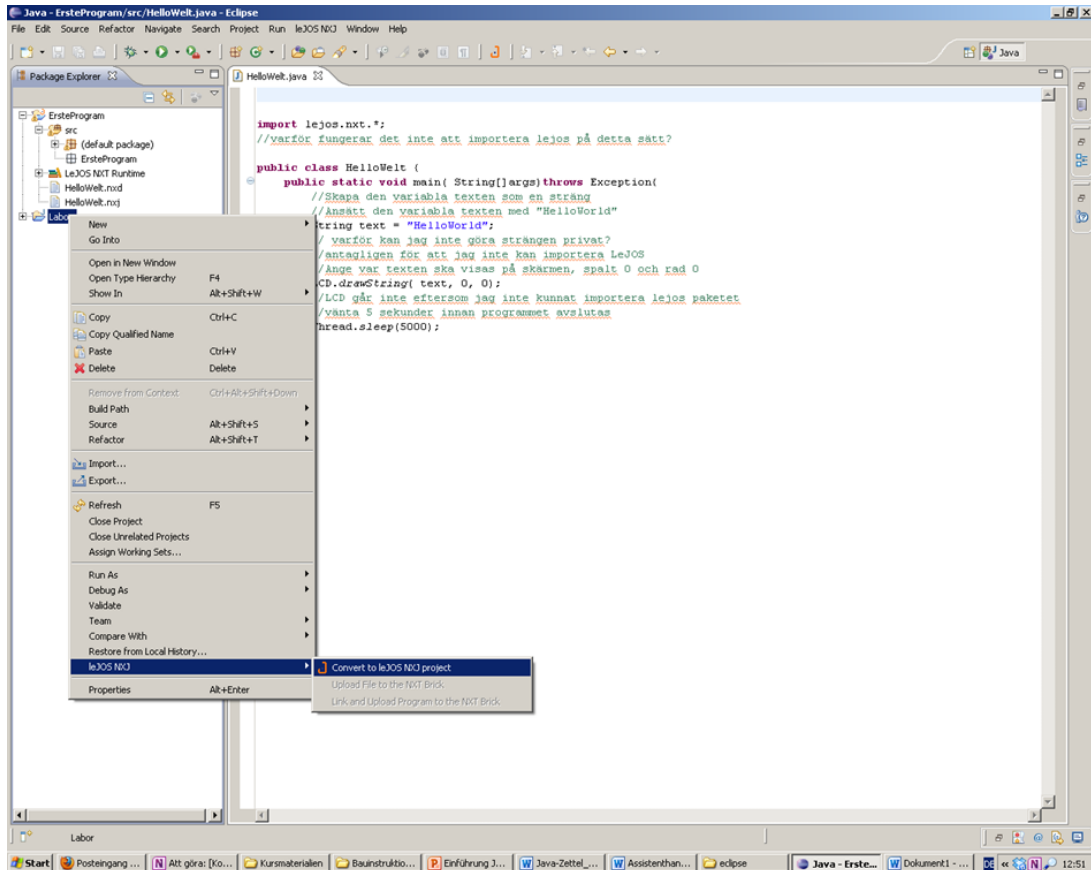
1. Börja med att öppna **Eclipse** (ligger på skrivbordet).
2. Sedan skapar vi ett nytt projekt. **File** → **New** → **Java Project**



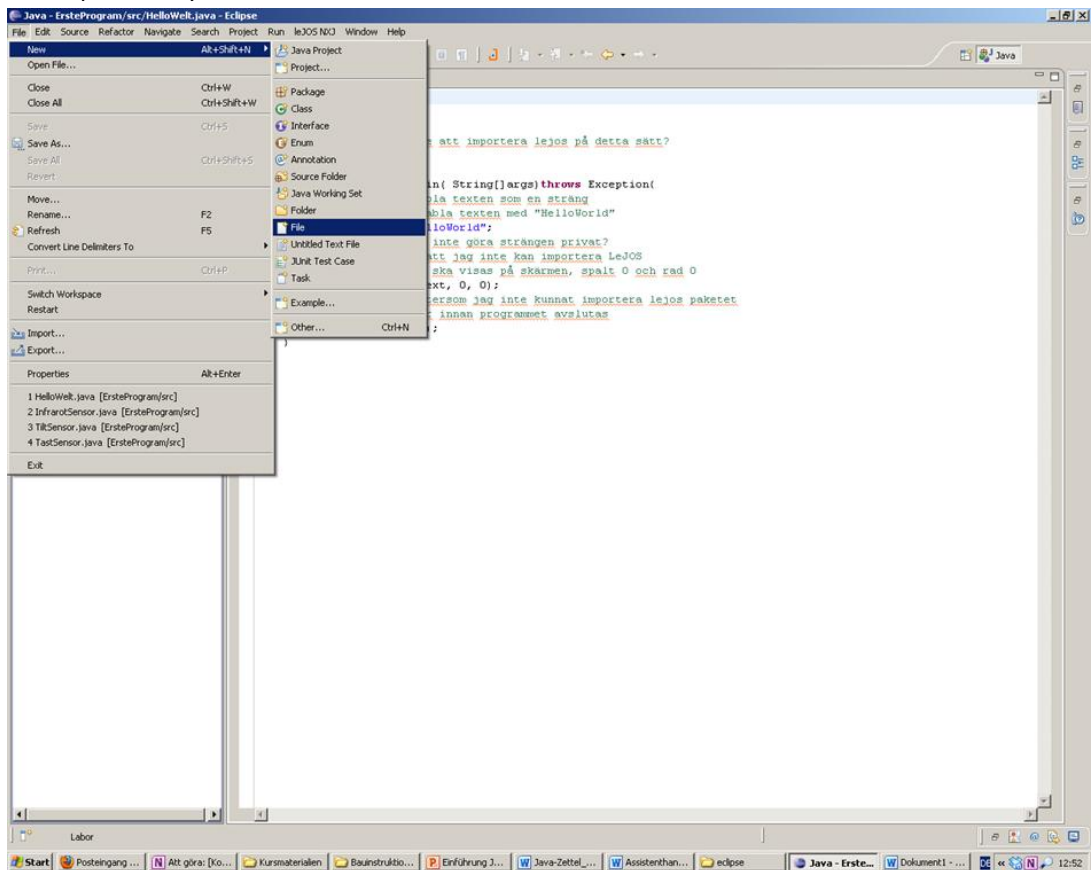
3. Ge det nya projektet ett namn och tryck sedan **Finish**.



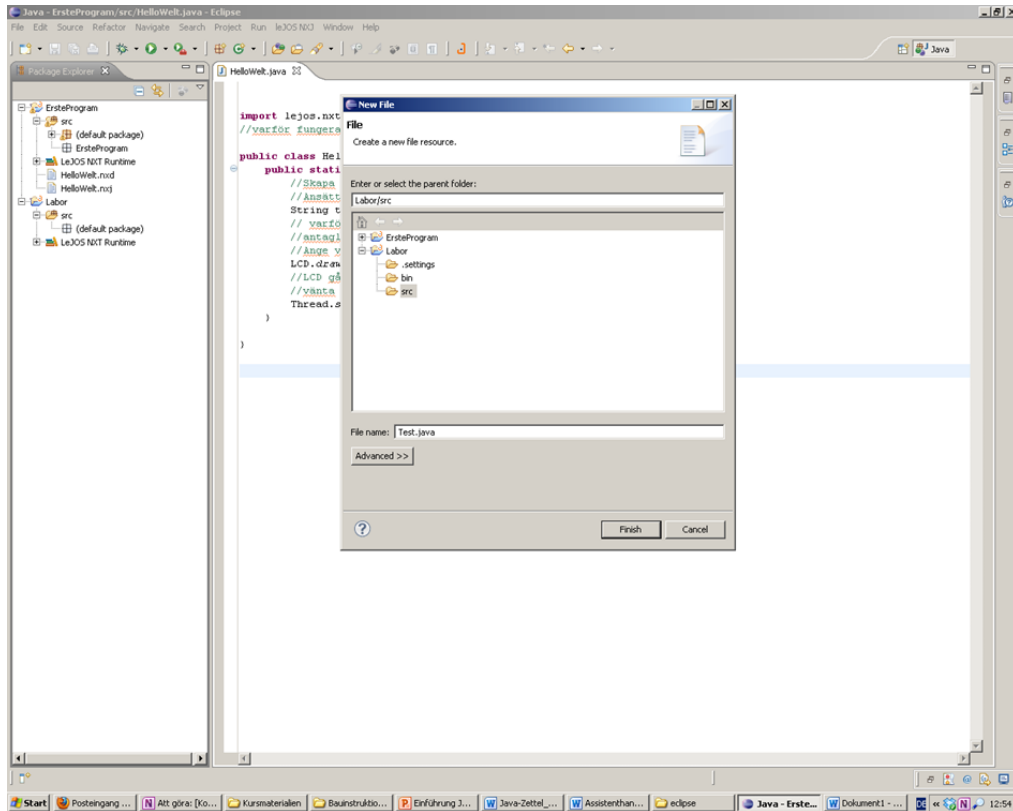
4. Konvertera projekt till Lejos Projekt. **Högerklicka å musen** → leJOS NXJ → **Convert to NXJ project**.



5. Nu skapar v en ny fil. **New** → **File**



6. Lägg filen i ditt projekt och *src*. Ge din fil ett namn *xxx.java*. Och tryck sedan *Finish*.



## Struktur av ett program:

```
import lejos.nxt.*;
// Importerar de viktigaste Lejos biblioteken.

public class xxx{
    public static void main( String[]args) throws Exception{
        ...
        ...
    }
}
```

### Slingor:

```
while (krav) {...}
```

```
for( int i = 0; i < 10; i++){...}
```

```
for( Deklaration; krav; förändring){...}
```

### Iteration:

```
if (krav) {...}
```

```
else{...}
```

## Olika funktioner:

### Displayen

På displayen kan både textmeddelanden och mätvärden/sensornvärden visas. På varje rad kan vi skriva 16 tecken (0 till 15), och det finns plats för 8 rader (0 till 7). Punkten (0,0) befinner sig i det övre vänstra hörnet. Den klass displayen använder är LCD.



### Klassen LCD

Metod	Beskrivning
<code>drawString(String text, int x, int y)</code>	Ger en text på bildskärmen. Parametern för denna text anges med, x för spalten och y för raden.
<code>drawString(int tal, int x, int y)</code>	Skriver ut ett heltal på skärmen. Parametern tal är för tal, x definierar spalten och y raden.
<code>Clear()</code>	Denna funktion tömmer displayen och behöver inga parametrar.

## NXT-motor

Motorerna har en maximalvinkelhastighet på 900° per sekund, vilket motsvarar ungefär 2,5 varv på sekund. Motorerna kan styras med så små marginaler som 1°. Då motorerna är servomotorer har de interna rotationssensorer som mäter hur många grader motorn roteras. Detta gör det möjligt att beräkna hur lång sträcka roboten färdats. På detta sätt kan även motorerna användas som sensorer.



### Klassen Motor

Metod	Beskrivning
<code>setSpeed(int gps)</code>	Sätter motorhastigheten till grad/sekund (gps), där 900 är maximum.
<code>setPower(int kraft)</code>	Anger motorkraften och ska vara ett värde mellan 0 och 100.
<code>Forward()</code>	Låter motor köra konstant framåt.
<code>Backward()</code>	Kör motorn konstant bakåt.
<code>Stop()</code>	Orsak till ett omedelbart stop och håller motorn på position (låser motorn).. Avslutar t.ex. <i>forward()</i> funktionen..
<code>rotate(int vinkel)</code>	Roterar motorn tills vinkeln <i>vinkel</i> är nådd, sedan stängs motorn av.

## Ultraljudssensorn

Denna sensor mäter avståndet till olika objekt (i cm). Den kan "se" kantiga föremål som befinner sig inom ca 30° (-15° bis + 15°) framför sensorn. Det maximala avståndet sensorn kan mäta är ca 250 cm, men det får fort stora mätfel. Vid ett avstånd på ca 150 cm kan sensorn mäta avståndet till föremål inom 20°. Minskar avståndet mellan sensorn och föremålet upptäcks även roterande objekt.



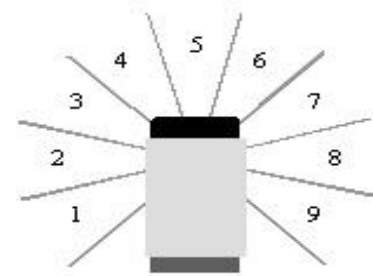
```
UltrasonicSensor ii = new UltrasonicSensor(SensorPort.S1);  
Ersätter Ultraschallsensor med ii.
```

### Klassen UltrasonicSensor

Metod	Beskrivning
<code>Int getDistance()</code>	Ger avståndet till funna objekt tillbaka.

## Infrarödsensor

Denna sensor mäter avståndet till en infrarödkälla. IRSeeker använder en digital signal och filtrerar bort bakgrundssignaler. Med IRSeeker kan vi få ett värde (se bilden bredvid) eller rader (mellan -180 och 180) för att se åt vilket håll källan befinner sig. Den kan även mäta styrkan.



```
import lejos.nxt.addon.IRSeekerV2;  
import lejos.nxt.addon.IRSeekerV2.Mode;  
Importerar biblioteken för infrarödsensorn.  
IRSeekerV2 iii = new IRSeekerV2(SensorPort.S2,  
Mode.AC);  
Ger den infrarödsensorn namnet iii.  
iii.setAddress(0x10);  
Nödvändiga inställningar för sensorn.
```



### Klassen IRSeekerV2

Metod	Beskrivning
Float getAngle(boolean blocking)	Ger vinkeln till målet (mellan -180 och 180 grader).
Int getDirection()	Ger riktningen till det föremålet med högst infrarött värde (mellan 1 och 9).
Int getSensorValue(int id)	Ger intensiteten av strålningen, Sensorerna är numrerade 1-5 och ligger i platserna 1,3,5,7 och 9 på bilden.

# Fragebogen

Bitte beantworte die folgenden Fragen so ehrlich wie möglich.

1. Ich bin:  Junge  Mädchen
  2. Hast du schon früher Erfahrungen mit Robotern gemacht?  
 Ja  Nein
  3. a. Hast du schon vorher Erfahrungen mit Programmieren gemacht?  
 Ja  Nein  
b. Wenn ja, welche Programmiersprache hast du früher schon verwendet?
- 

Bitte gib bei den folgenden Fragen an, wie stark du den Aussagen zustimmen würdest. Wenn du die Zahl 1 ankreuzt, stimmst du der Aussage sehr stark zu. Wenn du die Zahl 6 ankreuzt, stimmst du der Aussage überhaupt nicht zu.

4. Der Kurs macht mir Spaß.  
 1  2  3  4  5  6
5. Der Kurs war interessant.  
 1  2  3  4  5  6
6. Ich fand den Kurs heute allgemein gut.  
 1  2  3  4  5  6
7. Der Assistent hat eine gute Arbeit gemacht.  
 1  2  3  4  5  6
8. Der Kurs hat bei mir Interesse für die Robotik geweckt.  
 1  2  3  4  5  6
9. Der Kurs hat bei mir Interesse für die Informatik geweckt.  
 1  2  3  4  5  6
10. Ich würde gerne nochmal einen Kurs besuchen.  
 1  2  3  4  5  6
11. Der Kurs hat mir neue nützliche Kenntnisse vermittelt.  
 1  2  3  4  5  6
12. Ich habe etwas gelernt.  
 1  2  3  4  5  6
13. Welche Teile des Kurses haben dir gefallen:  
a. Die Demostraße hat mir gut gefallen.  
 1  2  3  4  5  6



b. Die Theorie-Vorträge über Robotik und Programmierung haben mir gut gefallen.

1 2 3 4 5 6

c. Die Bauphase und Konstruktionsanleitung haben mir gut gefallen.

1 2 3 4 5 6

d. Die Programmierphase hat mir gut gefallen.

1 2 3 4 5 6

e. Die Materialien (Handzettel, SMART-board, Präsentationen, etc.) haben mir gut gefallen.

1 2 3 4 5 6

f. Die Vorstellung des Roboters und der abschließende Vortrag haben mir gut gefallen.

1 2 3 4 5 6

g. Ich fand das Thema spannend.

1 2 3 4 5 6

h. Die Zeiteinteilung fand ich angemessen.

1 2 3 4 5 6

14. Welche Dinge findest du am Kurs gut?

---

---

---

15. Welchen Teil findest du am interessantesten?

---

---

---

16. Welche Dinge findest du am Kurs nicht gut?

---

---

---

17. Was würde bei dir mehr Interesse an den MINT-Fächern (Mathematik, Informatik, Naturwissenschaften, Technik) wecken?

---

---

---

---

18. Sonstige Anmerkungen

---

---

---

---

---

---

---

**Vielen Dank!**

# Frågeformulär

Var vänlig och besvara följande frågor så ärligt så möjligt.

19. Jag är:  Kille  Tjej
20. Har du tidigare erfarenheter av robotar?  
 Ja  Nej
21. a. Har du tidigare erfarenhet av programmering?  
 Ja  Nej
- b. Om ja, vilka programmeringsspråk har du använt?
- 

Var vänlig och besvara följande frågor utifrån hur mycket du håller med. Om du instämmer helt kryssa i 1, om du inte håller med överhuvudtaget kryssa i 6.

22. Laborationen var rolig.  
 1  2  3  4  5  6
23. Laborationen var intressant.  
 1  2  3  4  5  6
24. Jag tycker att laborationen var bra.  
 1  2  3  4  5  6
25. Assistenterna gjorde ett bra arbete.  
 1  2  3  4  5  6
26. Laborationen väckte mitt intresse för robotik.  
 1  2  3  4  5  6
27. Laborationen väckte mitt intresse för datakunskap.  
 1  2  3  4  5  6
28. Jag skulle gärna besöka en laboration igen.  
 1  2  3  4  5  6
29. Laborationen lärde mig något användbart.  
 1  2  3  4  5  6
30. Jag lärde mig någonting.  
 1  2  3  4  5  6
31. Vilka delar av laborationen tyckte du var bra:
- a. Demo-Ytan var bra.  
 1  2  3  4  5  6
- b. Teoriföreläsningsdelen om robotik och programmering var bra.  
 1  2  3  4  5  6

1 2 3 4 5 6

c. Byggfasen och konstruktionsinstruktionen var bra.

1 2 3 4 5 6

d. Programmeringsfasen var bra.

1 2 3 4 5 6

e. Materialet (Stenciler, SMART-board, presentationer osv.) var bra.

1 2 3 4 5 6

f. Framställningen av robotarna och det avslutande föredraget var bra.

1 2 3 4 5 6

g. Jag tycker temat var spännande.

1 2 3 4 5 6

h. Tidsuppdeleningen var bra.

1 2 3 4 5 6

32. Vilka delar av laborationen fann du bra?

---

---

---

33. Vilka delar av laborationen fann du mest intressanta?

---

---

---

34. Vilka delar av laborationen fann du inte bra?

---

---

---

35. Vad skulle intressera dig mer för MINT-ämnena (Matematik, Datakunskap, Naturvetenskap och Teknik)?

---

---

---

36. Övriga kommentarer

---

---

---

---

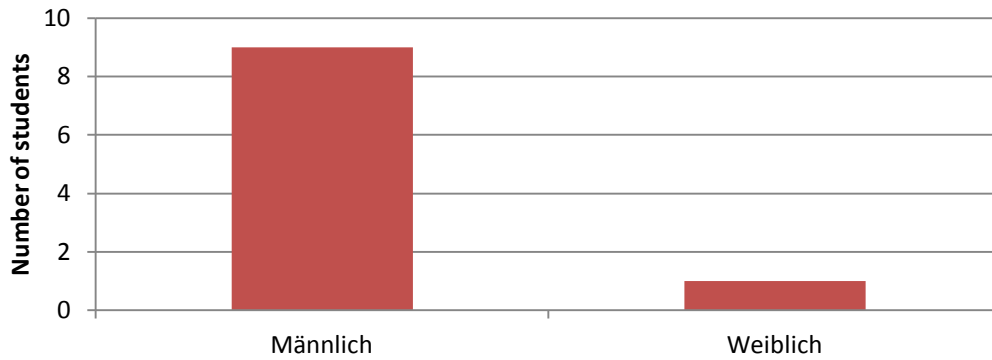
---

---

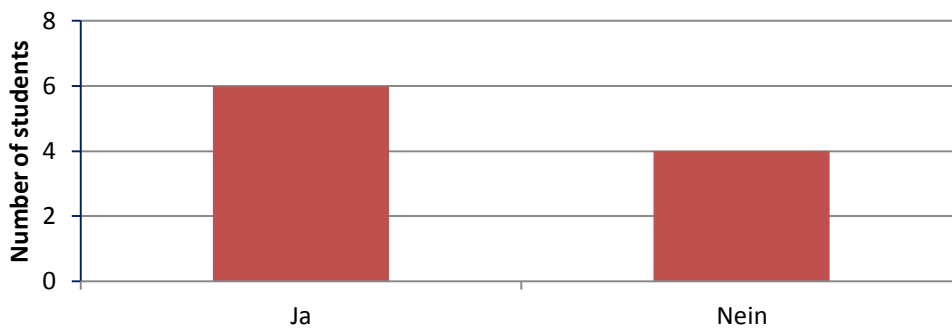
**Tack för hjälpen!**

## B Rückgabe von Fragebogen / Resultat av utvärderingen

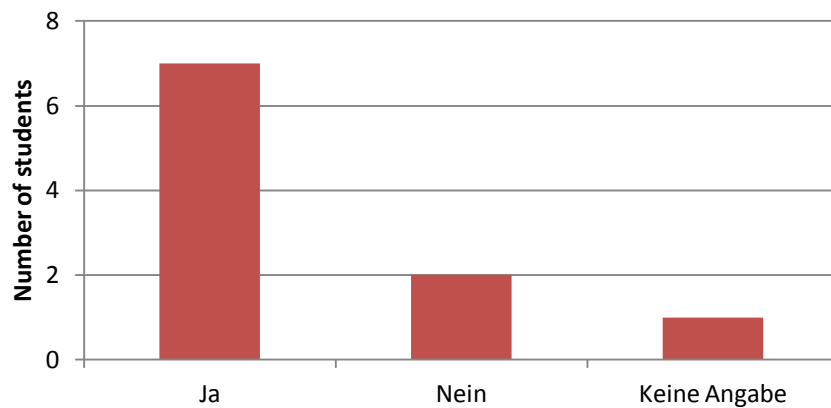
### Geschlecht



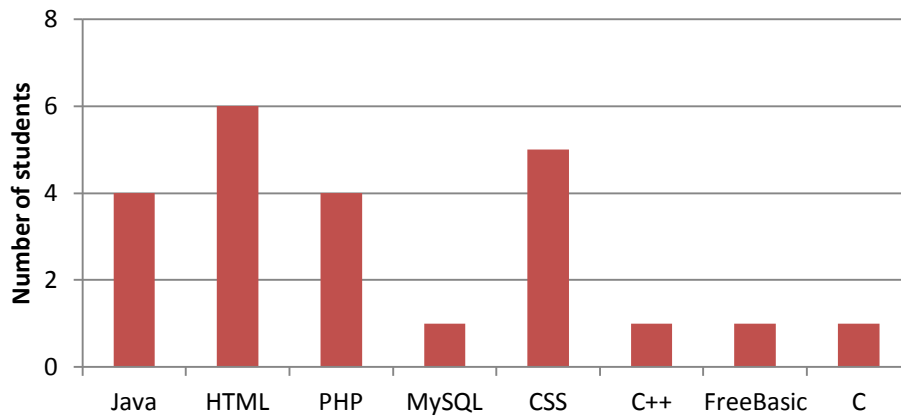
### Hast du schon früher Erfahrungen mit Robotern gemacht?



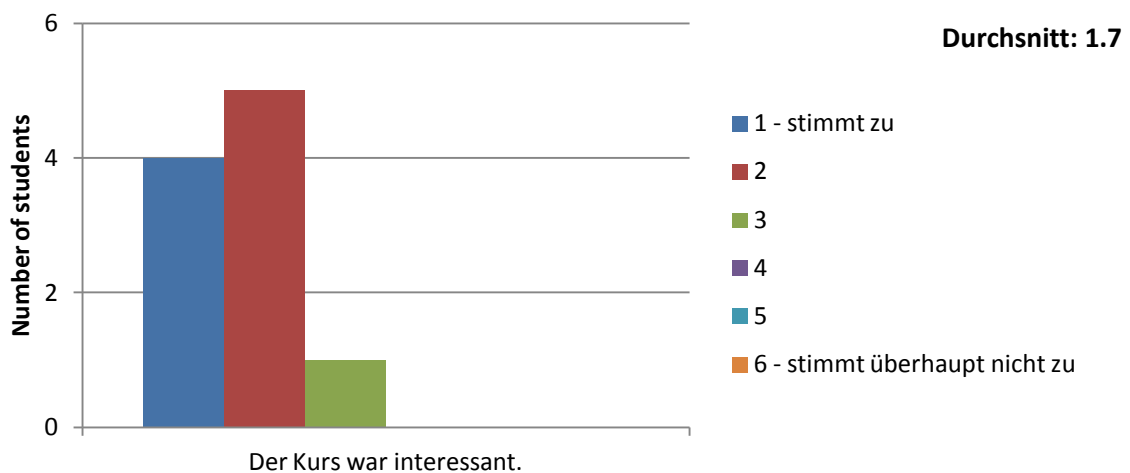
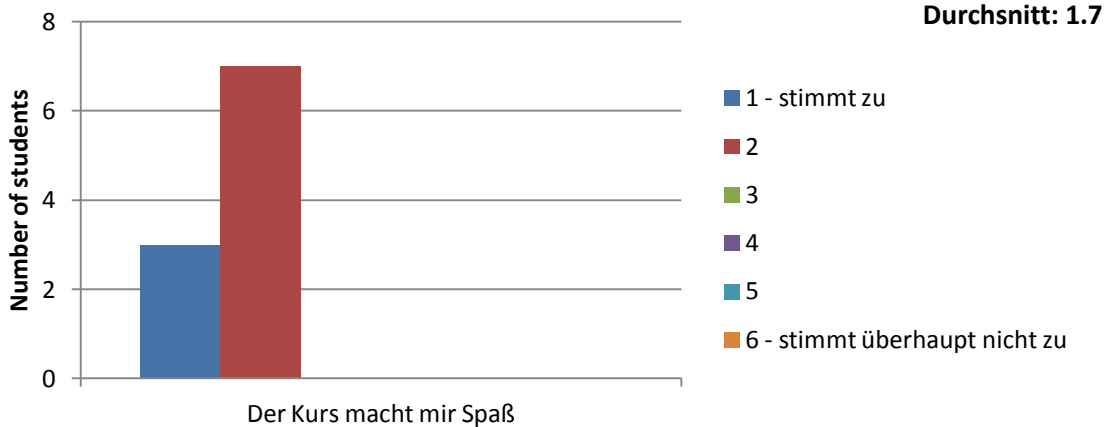
### Hast du schon vorher Erfahrungen mit Programmieren gemacht?

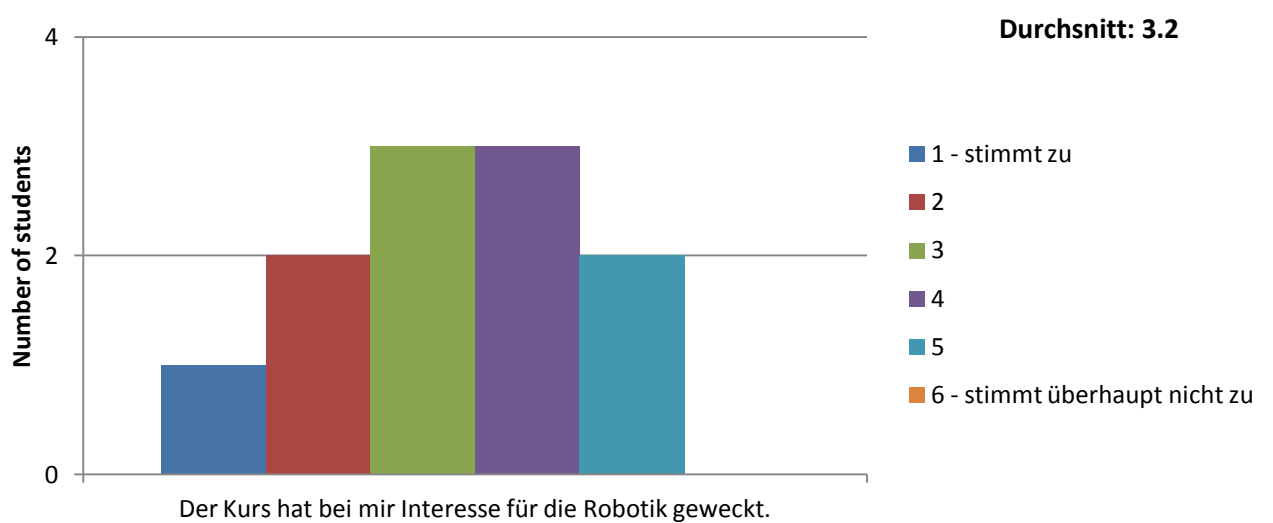
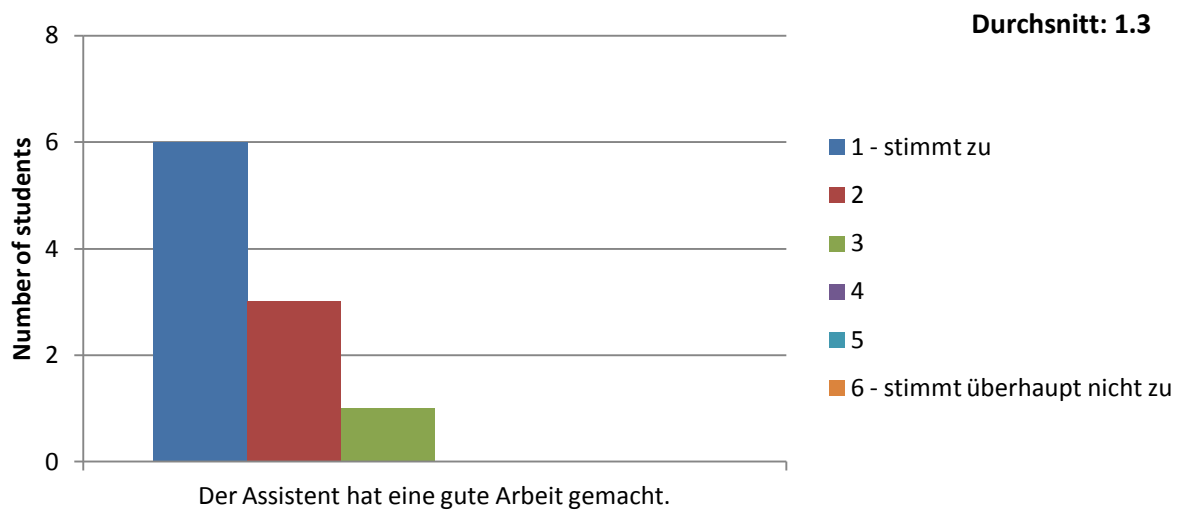
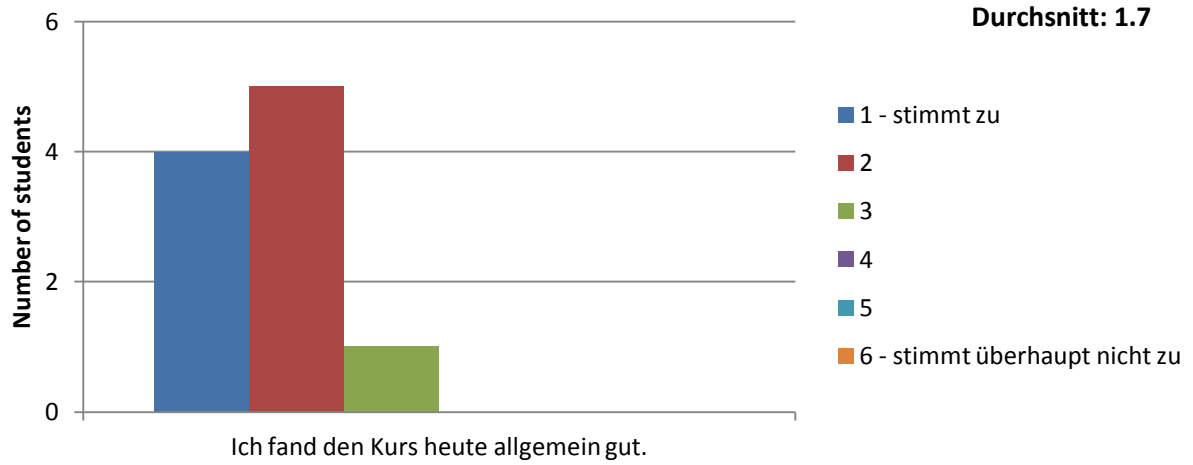


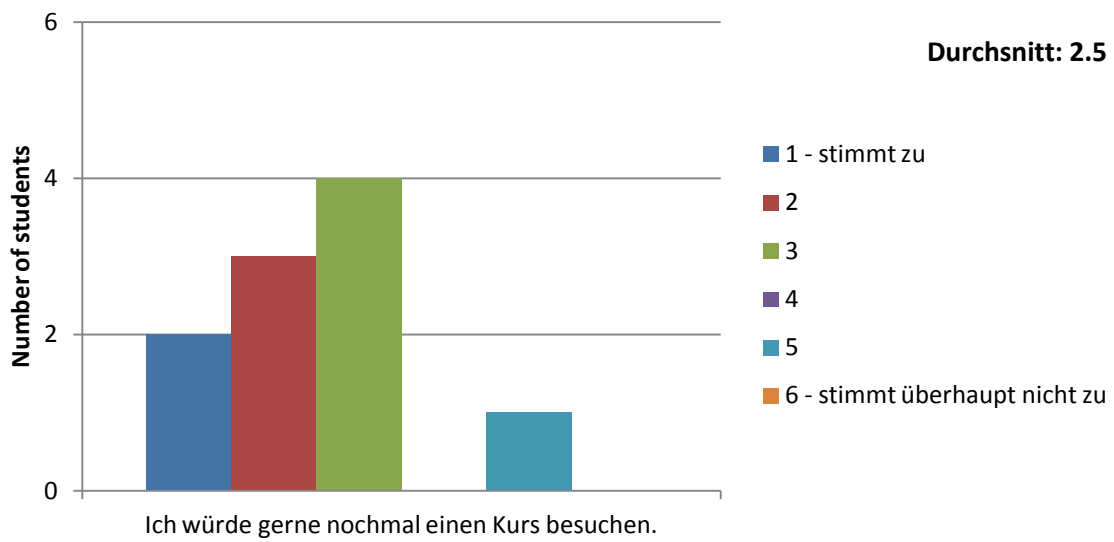
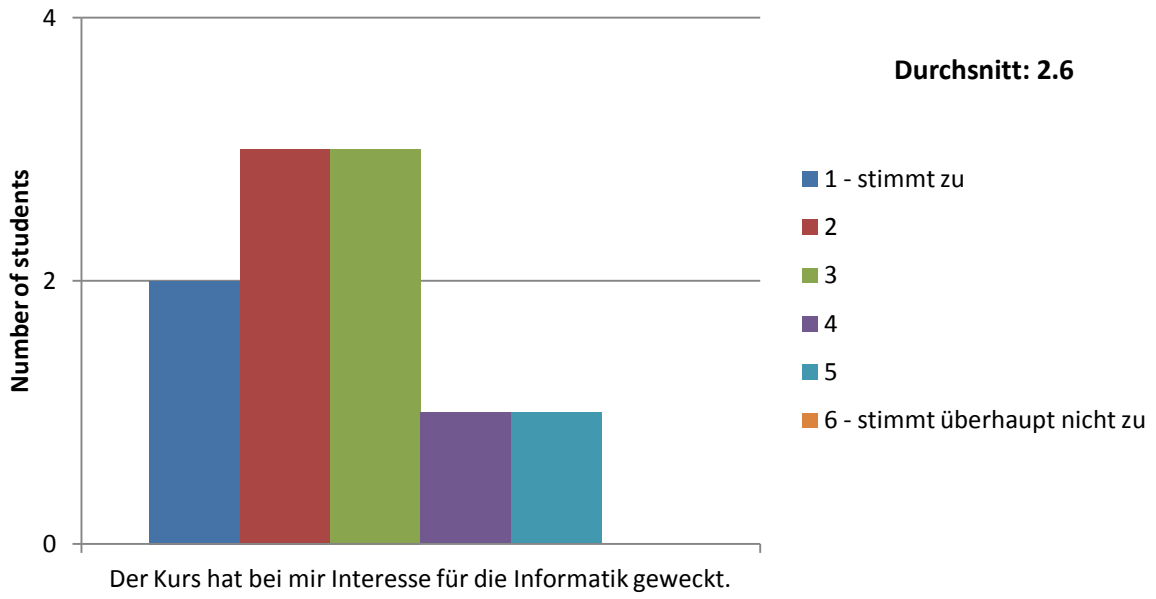
## Programmiersprache

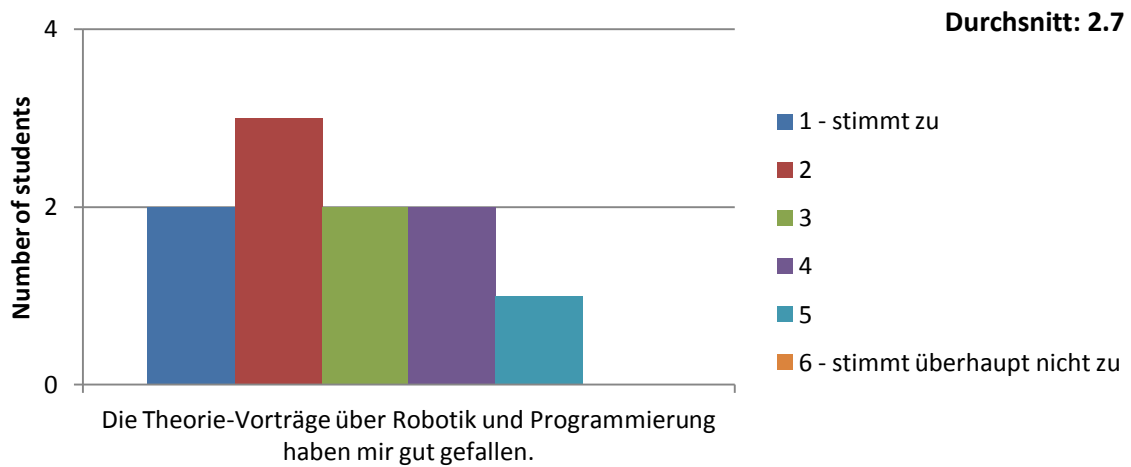
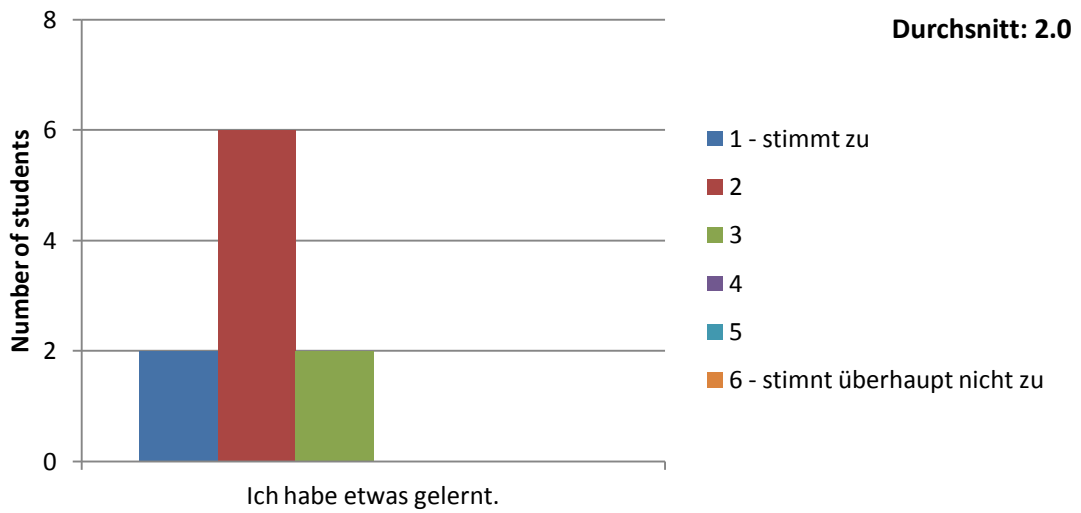
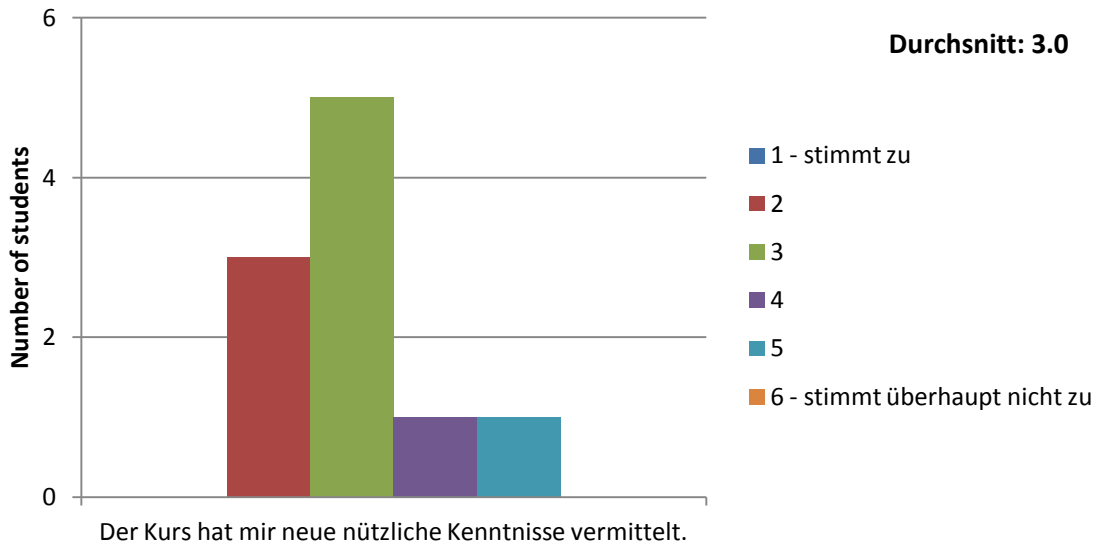


Beschreibung: Bitte gib bei den folgenden Fragen an, wie stark du den Aussagen zustimmen würdest. Wenn du die Zahl 1 ankreuzt, stimmst du der Aussage sehr stark zu. Wenn du die Zahl 6 ankreuzt, stimmst du der Aussage überhaupt nicht zu.

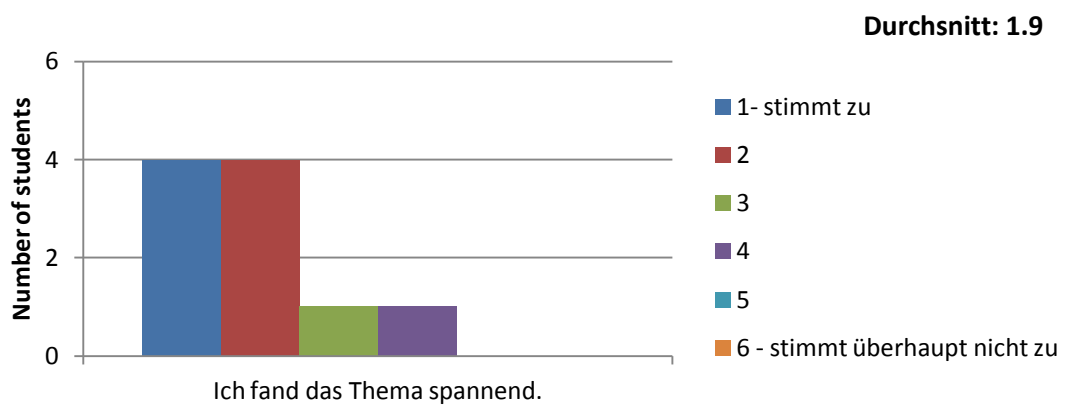
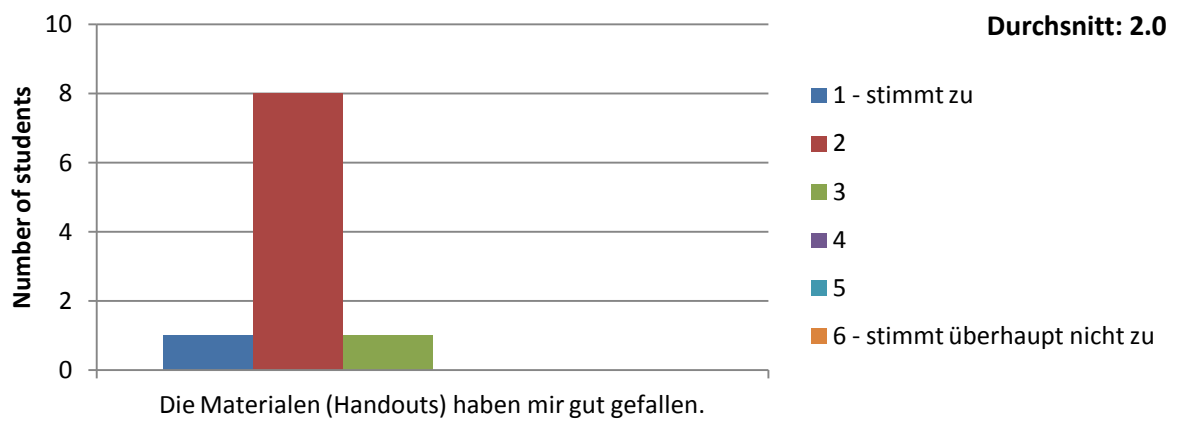
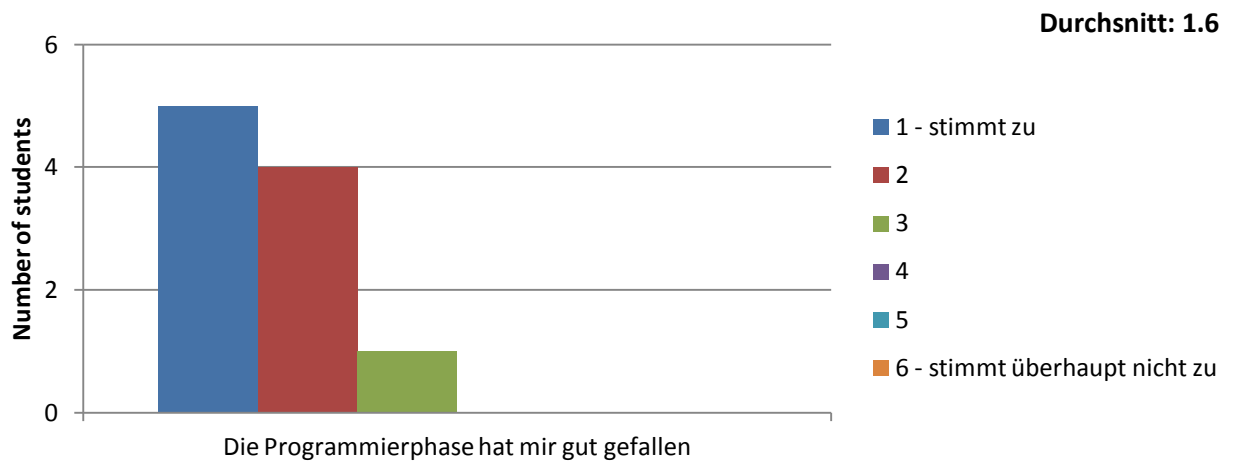




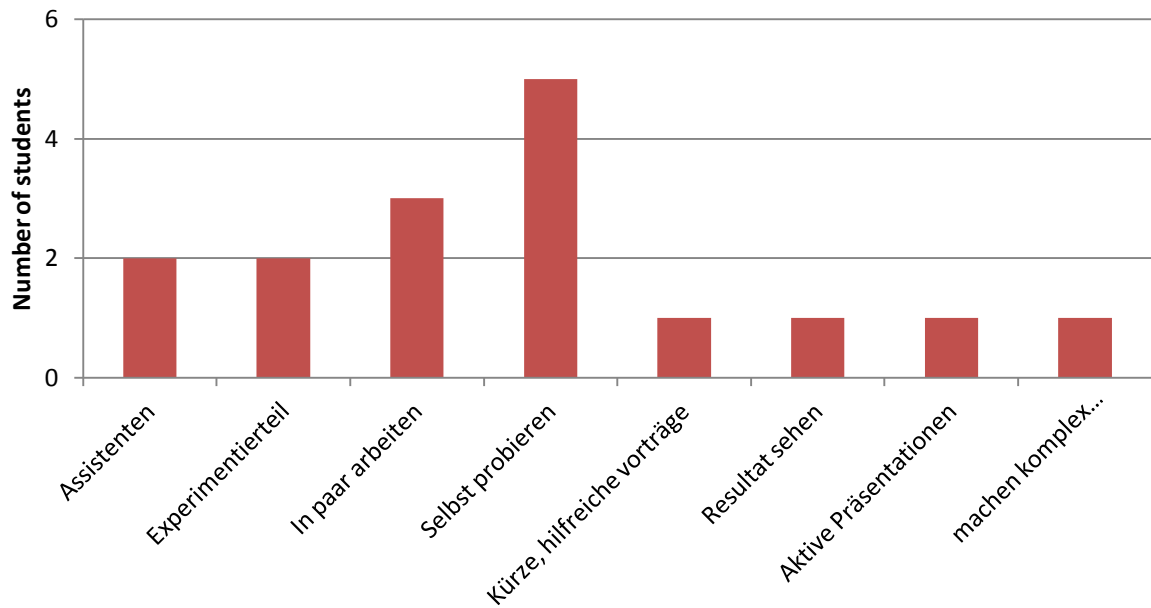




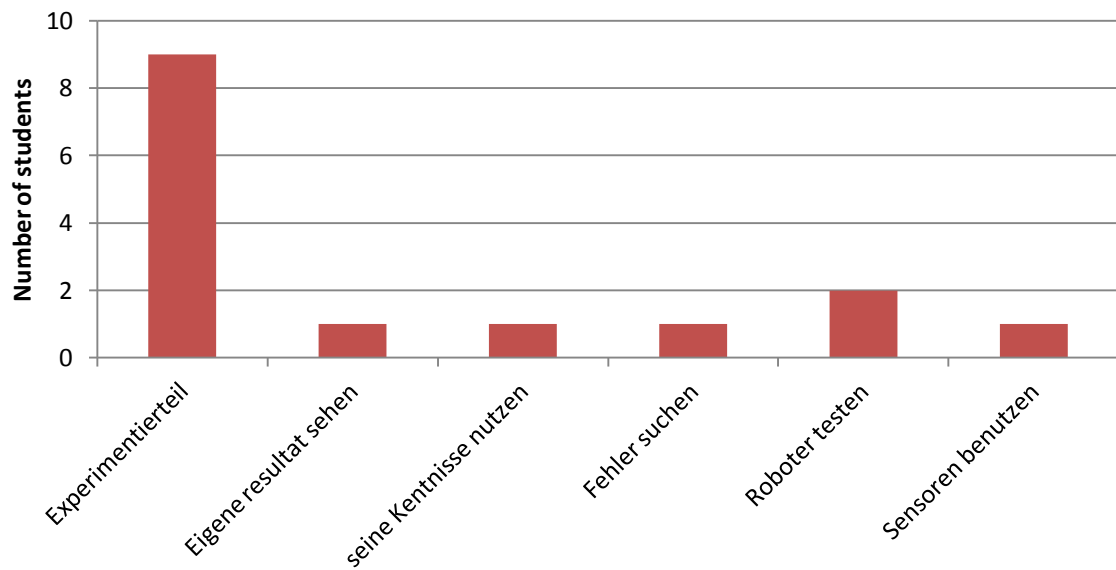




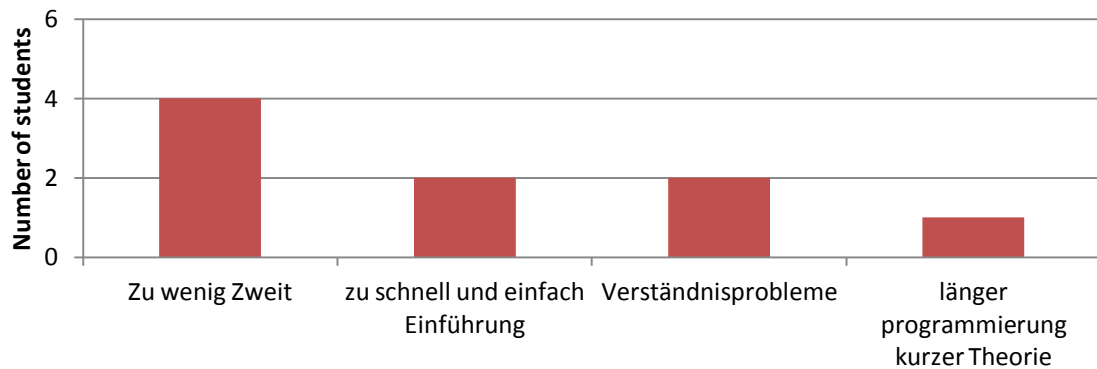
## Welche Dinge findest du am Kurs gut?



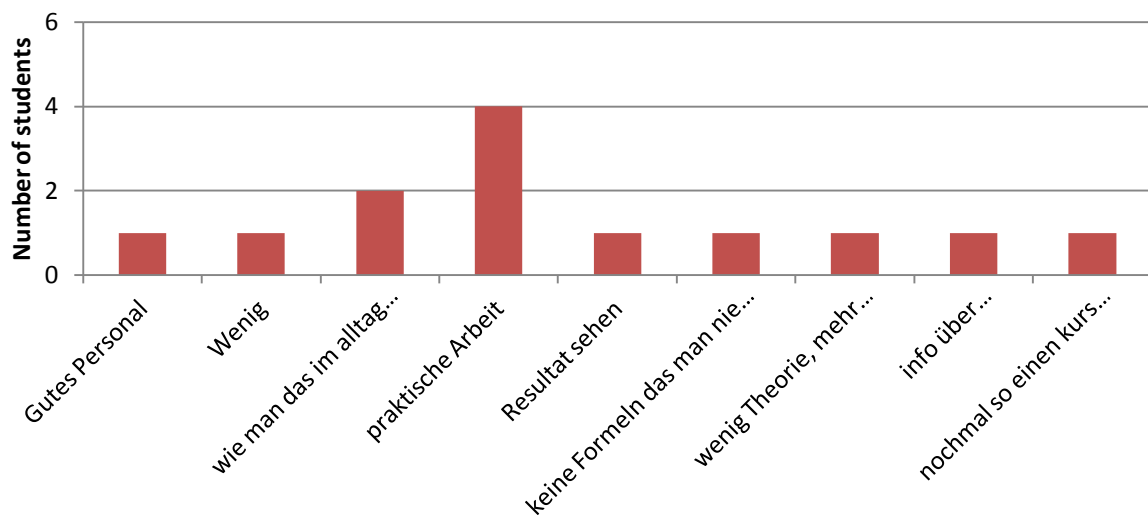
## Welchen Teil findest du am interessantesten?



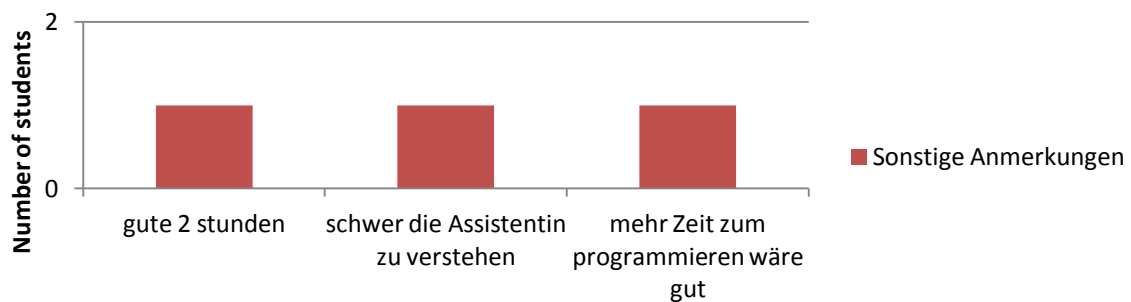
## Welche Dinge findest du am Kurs nicht gut?



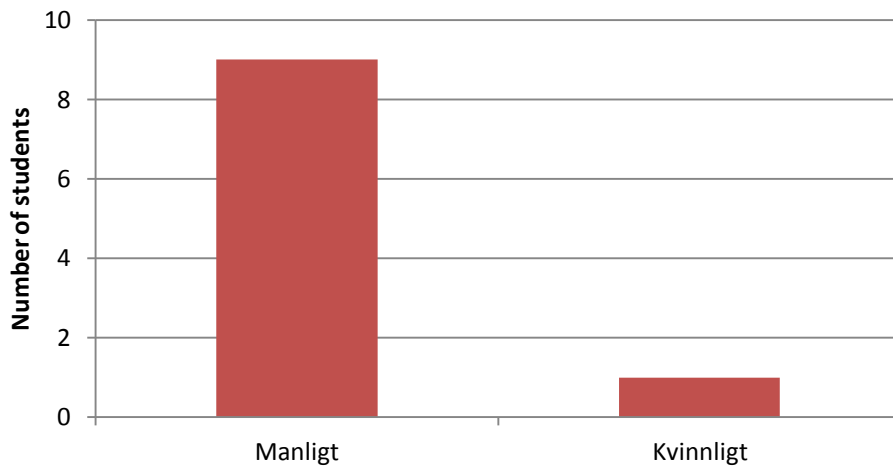
## Was würde bei dir mehr Interesse an den MINT-Fächern (Mathematik, Informatik, Naturwissenschaften, Technik) wecken?



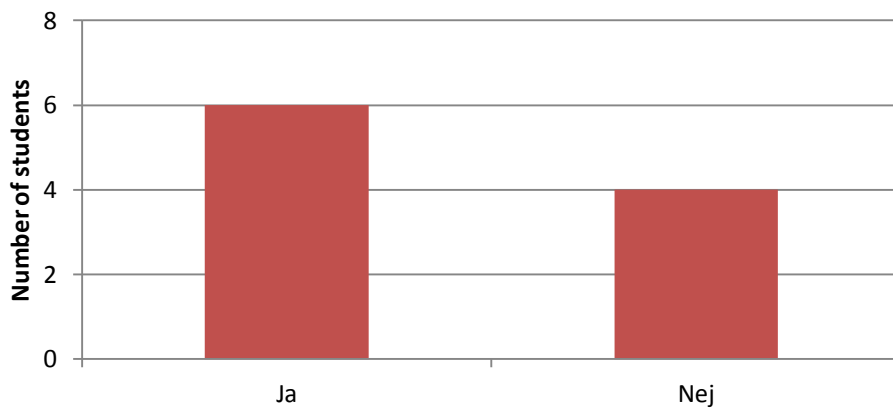
## Sonstige Anmerkungen



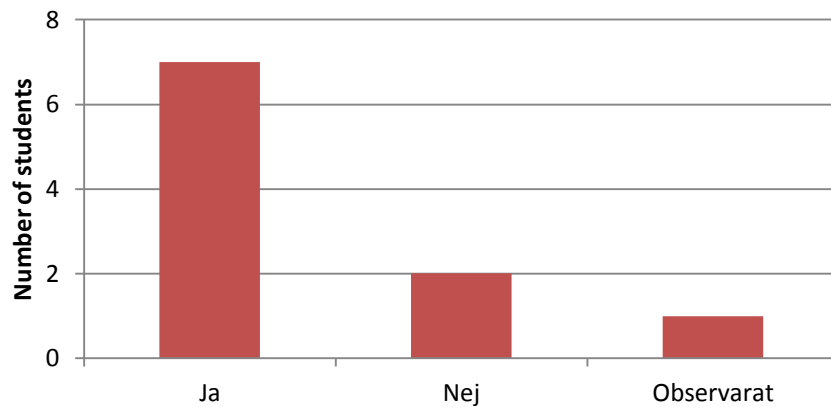
## Kön



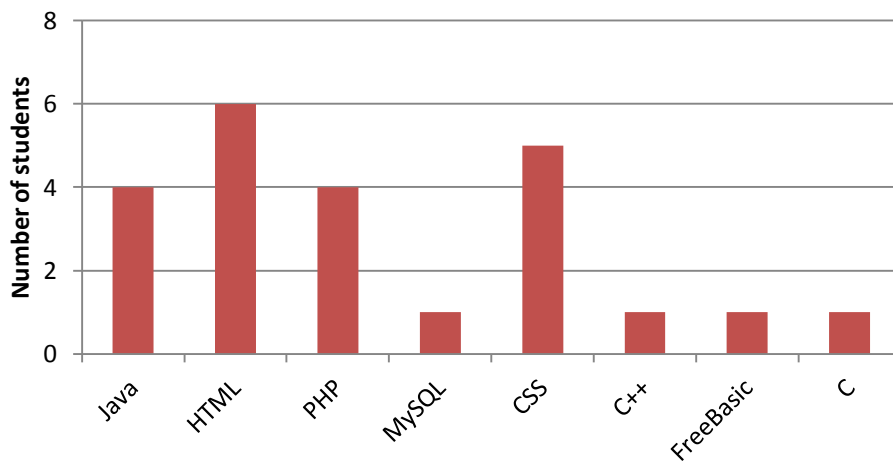
## Har du tidigare erfarenheter av robotar?



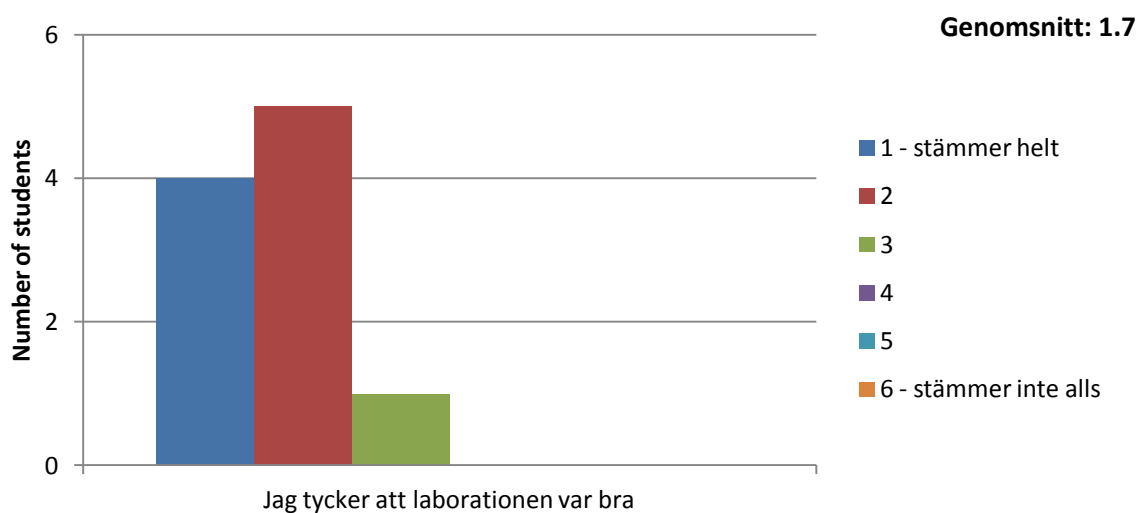
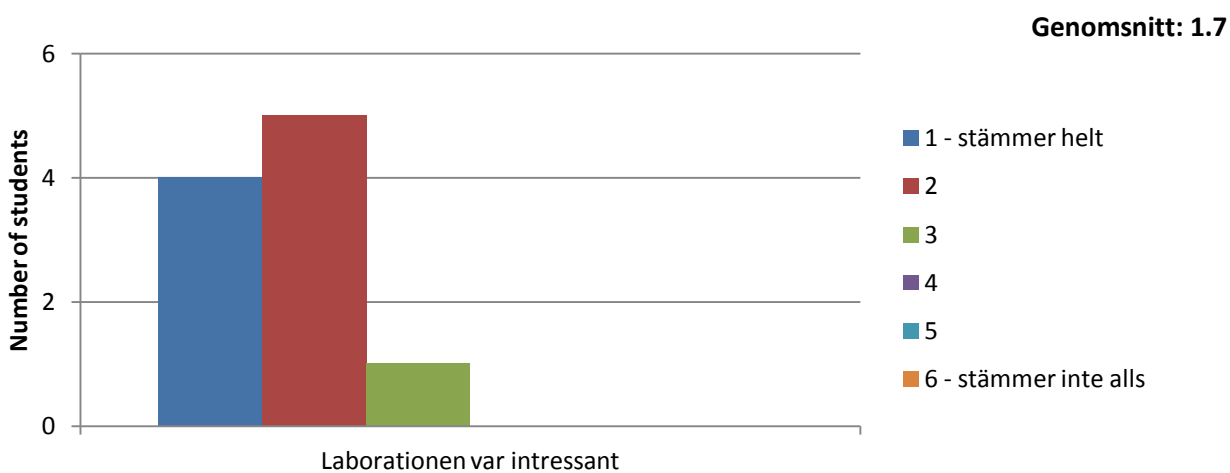
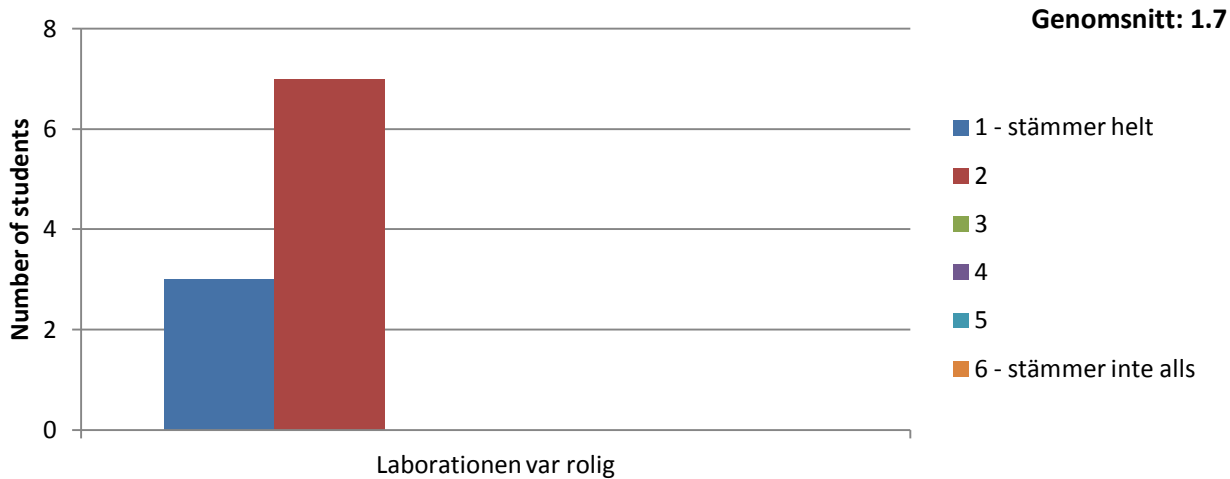
## Har du tidigare erfarenhet av programmering?

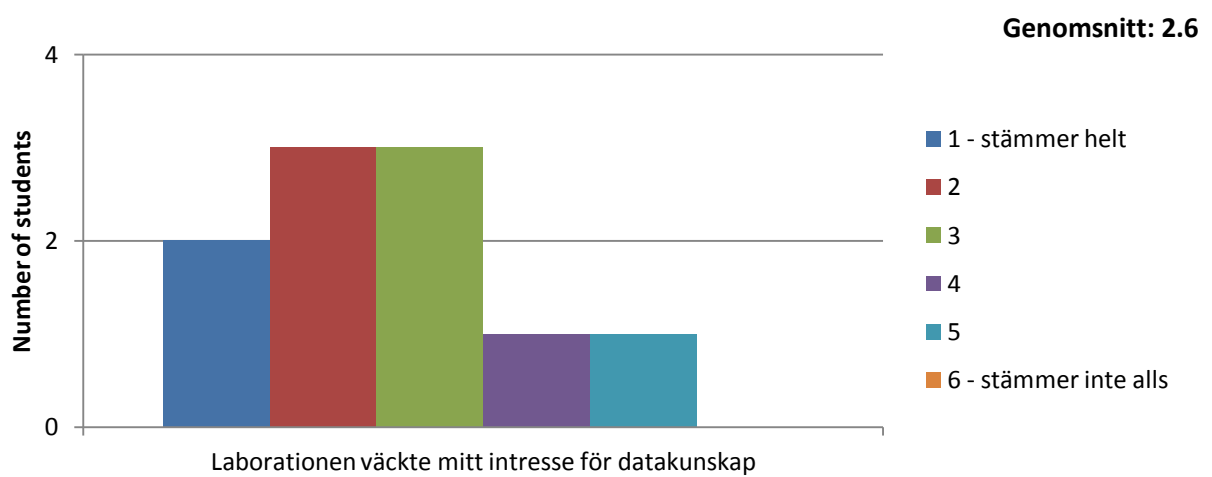
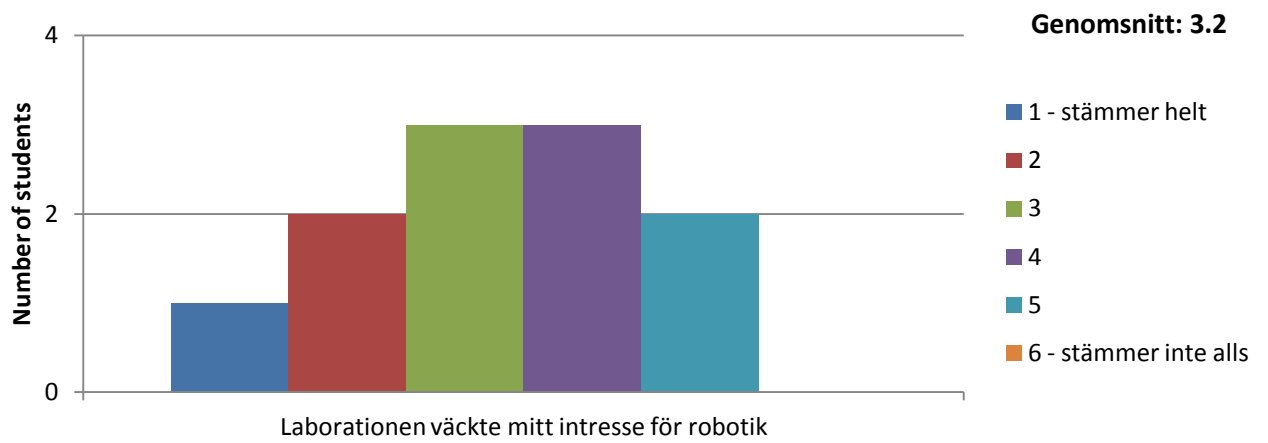
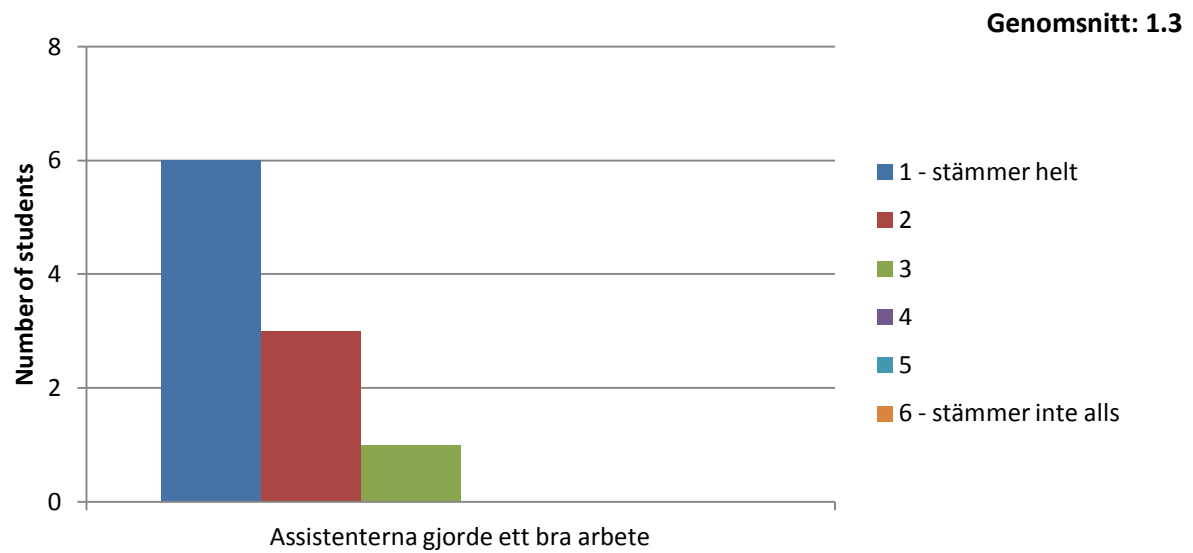


## Programmeringsspråk

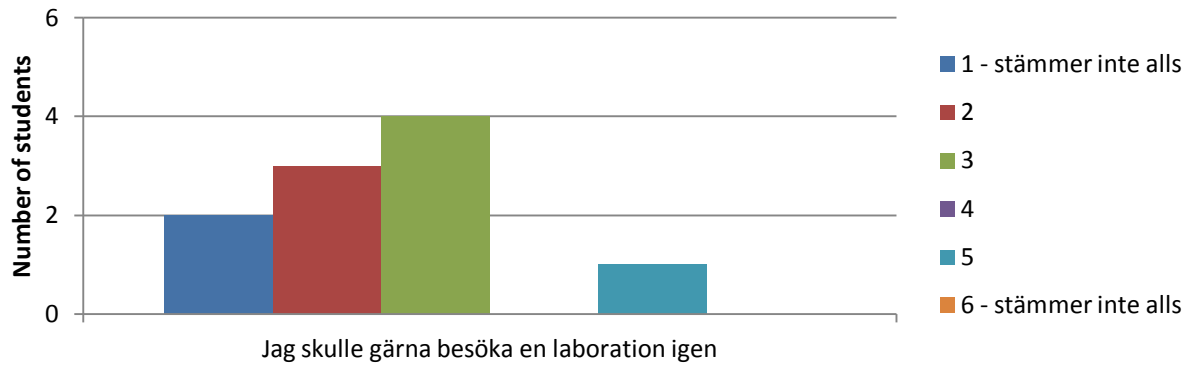


Beskrivning: Var vänlig och besvara följande frågor utifrån hur mycket du håller med. Om du instämmer helt kryssa i 1, om du inte håller med överhuvudtaget kryssa i 6.

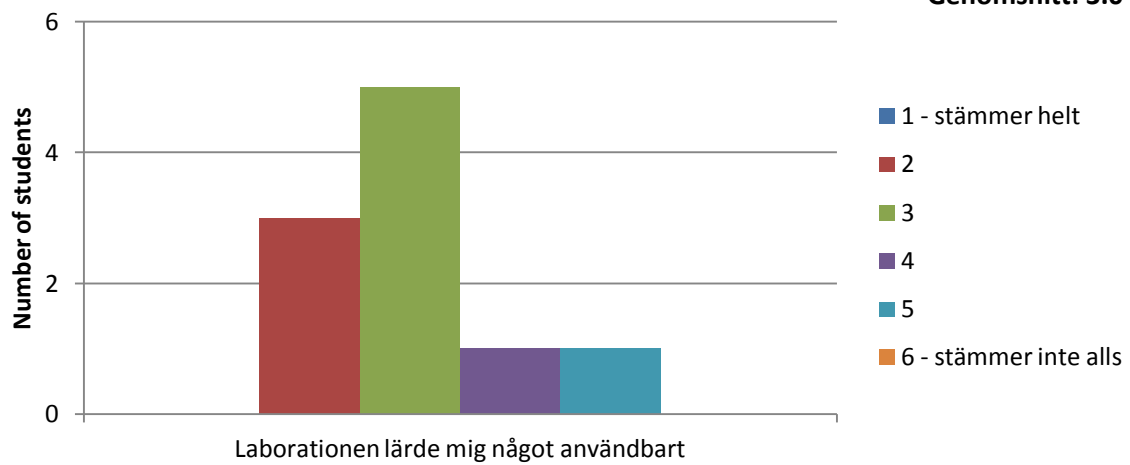




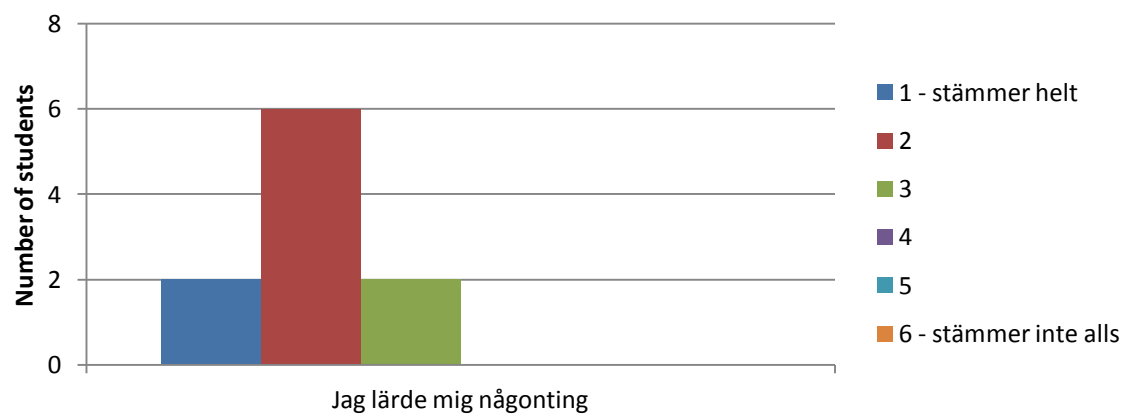
Genomsnitt: 2.5



Genomsnitt: 3.0

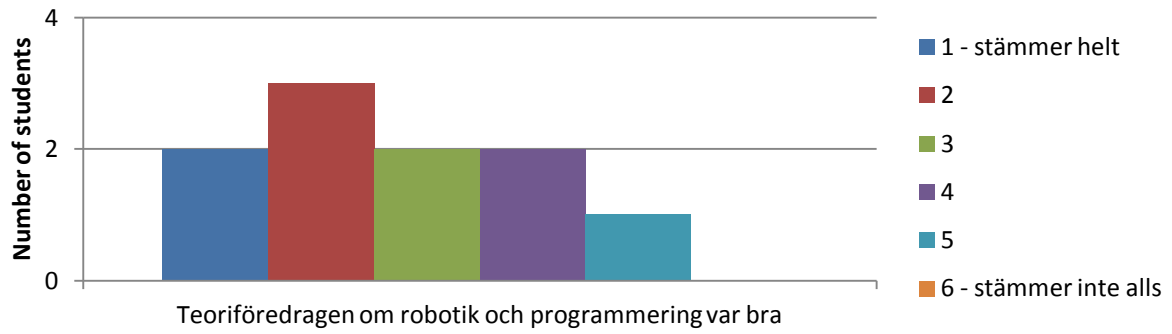


Genomsnitt: 2.0

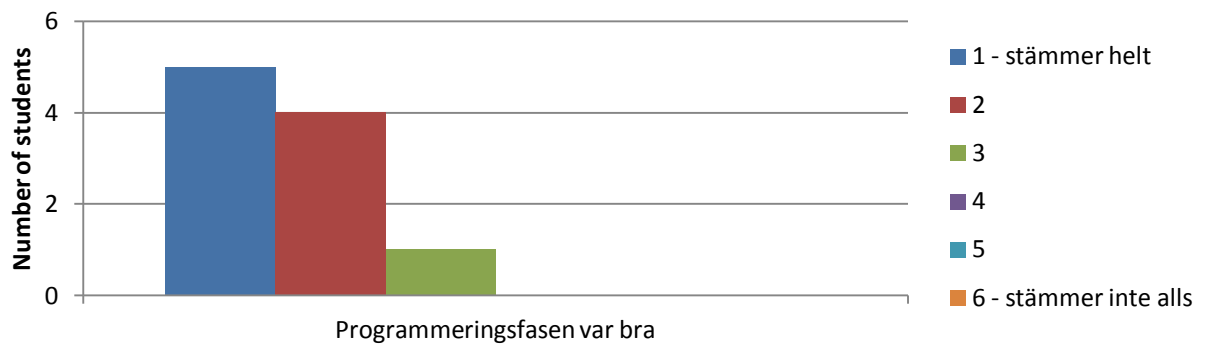




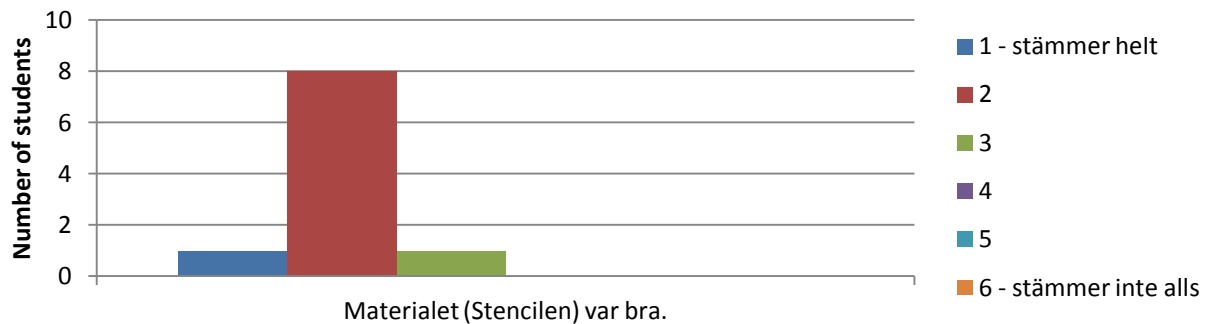
**Genomsnitt: 2.7**



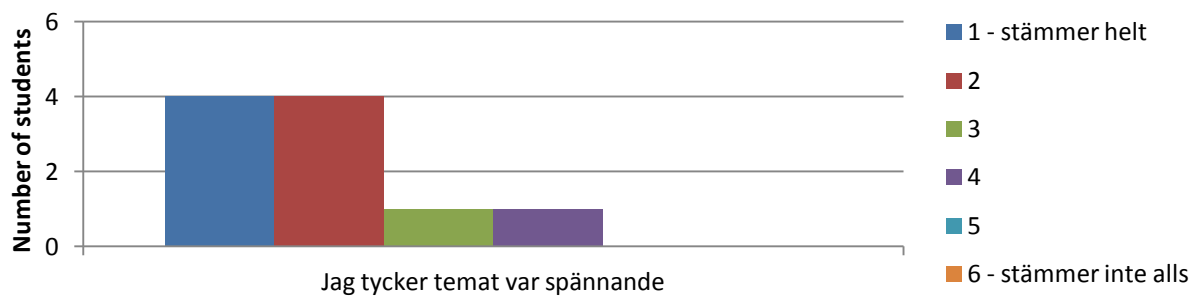
**Genomsnitt: 1.6**



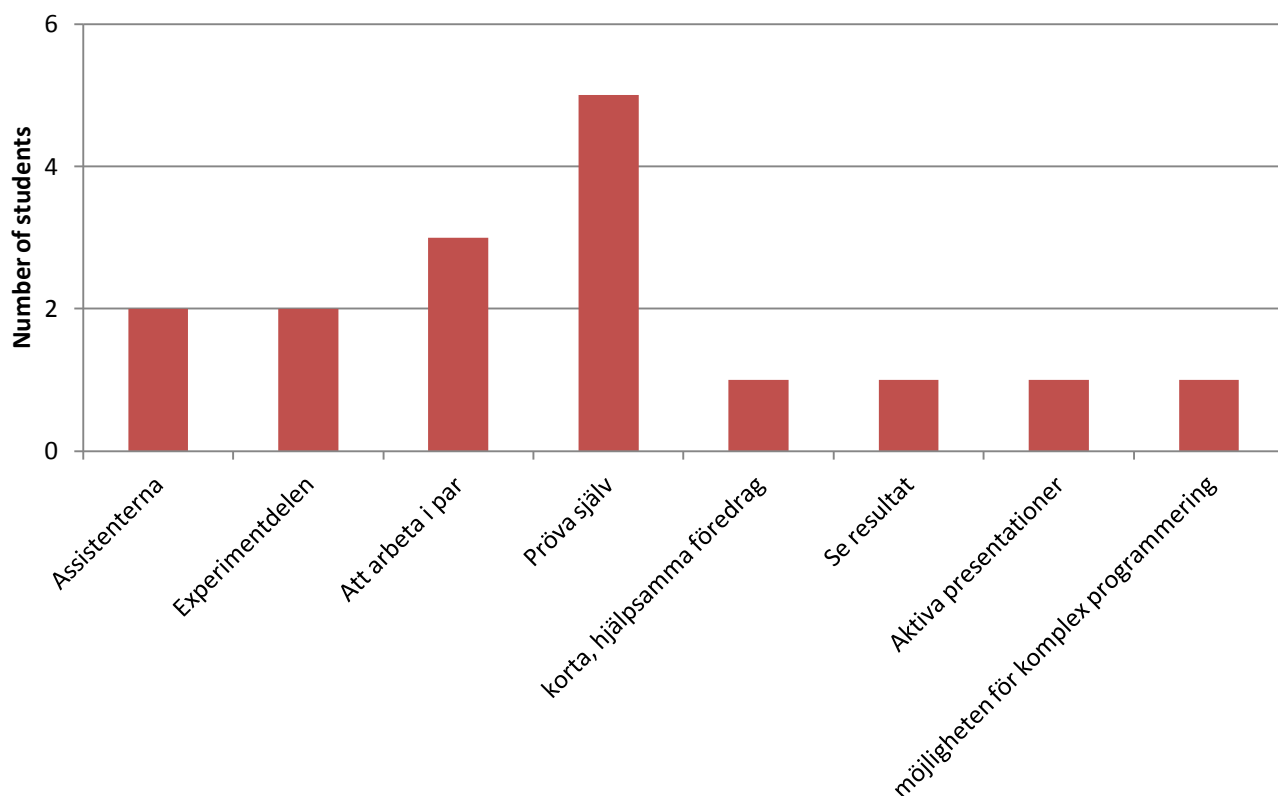
**Genomsnitt: 2.0**



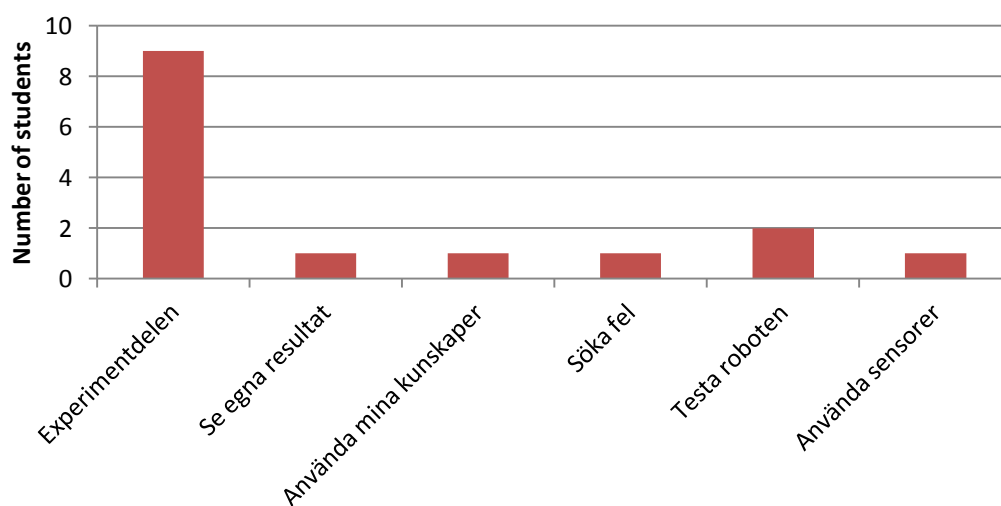
**Genomsnitt: 1.9**



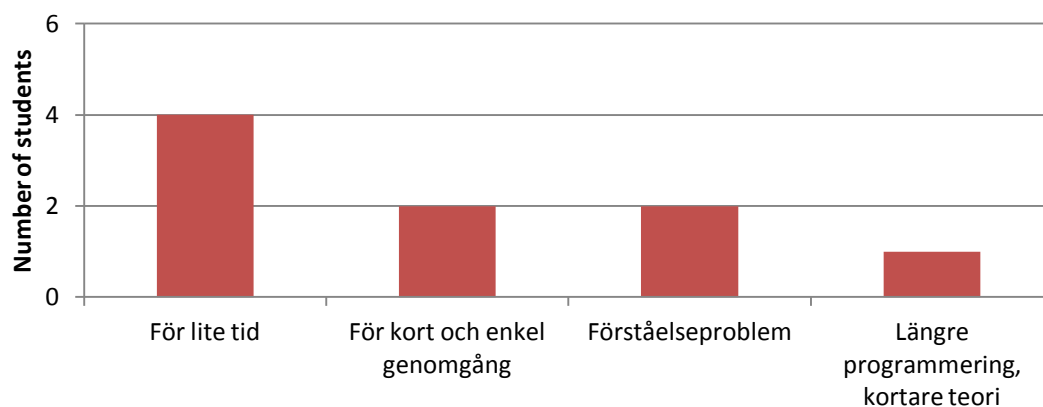
## Vilka delar av laborationen fann du bra?



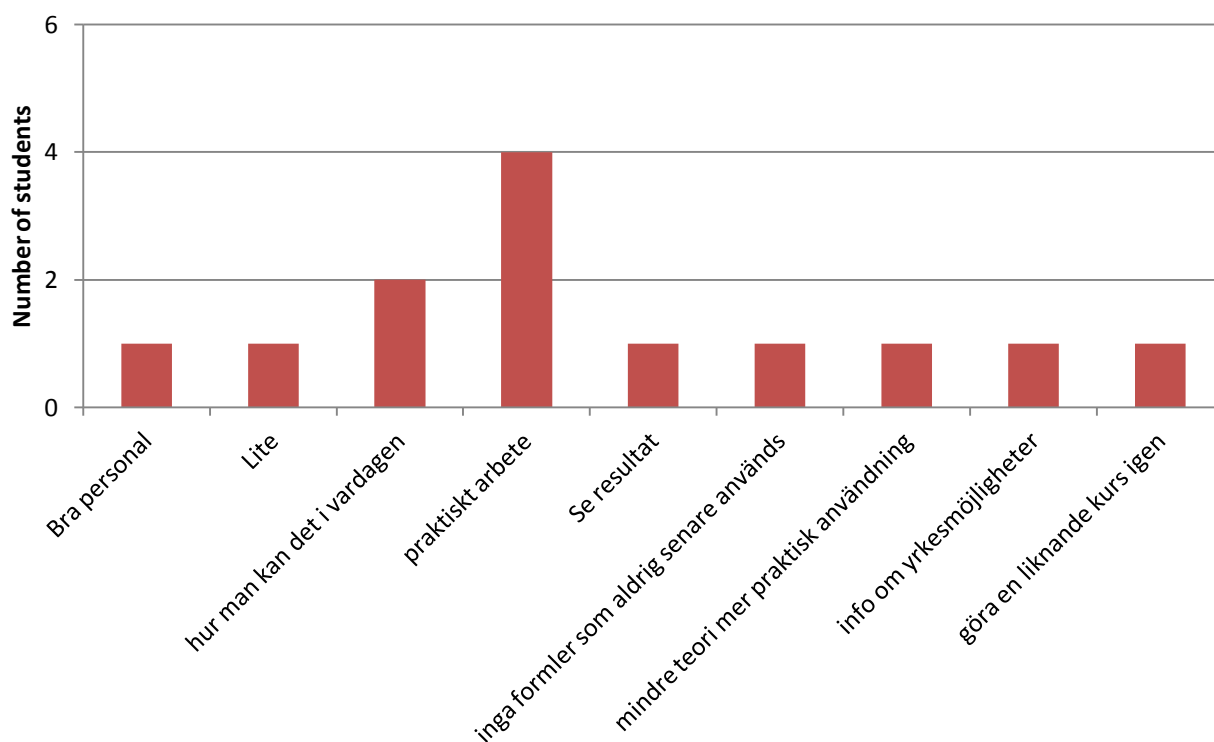
## Vilka delar av laborationen fann du mest intressanta?



## Vilka delar av laborationen fann du inte bra?



## Vad skulle intressera dig mer för MINT-ämnena (Matematik, Datakunskap, Naturvetenskap och Teknik)?



## Övriga kommentarer

