Course analysis (2016/17 period 1+2)
DD2425 Robotics and Autonomous Systems, 9hp

Course responsible and lecturer: Patric Jensfelt
Lab assistants: Diogo Almeida, Silvia Cruciani and Joshua Haustein
Number of lectures: 12 lectures (24 hours)
Registered students: 43 (according to VIS report VG.6)
Students passing: 35 (according to VIS report VG.6)
"Prestationsgrad": 84% (89% last year)
"Examinationgrad": 81% (87% last year)

Course material:

- Optional Book: "Introduction to Autonomous Mobile Robots" by Roland Siegwart and Illah R. Nourbakhsh published by the MIT Press.
- Lecture notes: Available for download from BILDA

Examination requirements:

- LAB1 0.5hp (during first period)
  - Basic ROS assignment
- TEN1 3hp (after course)
  - Written exam
- PRO1 5.5hp (whole second period)
  - Project work
  - Weekly status reports from the project group
  - Presentation
  - Reflection on the work of oneself
  - Reflection on group work
  - Project report

General description:
The course runs over two periods with the first period providing an overview of the field of robotics and the second being entirely focus on a project. The project starts about halfway into the first period. The project takes a lot of time.

The course evaluation was completed by 29 students after many reminders.

The course was appreciated by all students that answered the survey, 83% very interesting course and 14% interesting. This is MUCH higher than last year when 57% said very interesting and 43% interesting. The number of student that would recommend he course is 90% (66% very much so) compared to last years 71% this year.

Grading:
We use of criteria based system. It is still VERY challenging to set or rather

properly motivate the grades. One of the largest contributing factors for this is that most students spend very much time on the course. You can spend an enormous amount of time on the course if you do things wrong and are not good at debugging. It is therefore hard to separate such effort and performing well (and possibly spending much less time).

Relation to the previous years:

Project:

- Last year we aimed to give early feedback about grades but it was impossible because it was not until the very end when enough information was gathered about what actually went on in the systems to be able to make proper assessments. We tried to address that by have weekly meeting with each group. The idea was that we would be able to gather more information and also steer them better to avoid making costly errors. My impression is that this worked quite poorly from a grading point of view. It may even have skewed the picture in favor of those that can "sell themselves" in the presence of the examiner. It also consumed considerable amounts of time.
- We replace more or less the entire hardware setup. This was mostly successful.
- The biggest conceptual difference was that we added an arm to the robot and the task was changed to go in an pick up object. This made the task much less artificial.
- We changed the RGBD sensor from Primesense to an Intel RealSense SR300 sensor. This change was not so smooth as the Intel camera had some issues.

Planned changes:

- Work even hard to secure evidence of what they do and how earlier.
- Let most meeting be between group and TA and let these be more dynamic.
- Let the rules for the contest be known from the start. This has been avoided in the past as it has typically led to super specialized solutions based on the scoring function used. Maybe hold back on the scoring this year but make it more clear what problems they should aim to solve so that it is less searching in the unknown.

Survey results

Survey	Course evaluation
Event	DD2425 robot16
Status	open
Date	2017-04-19 18:10
Group	Participants
Answered by	29(43) (67%)

Provide a message to potential students for this course next year. Who should take it? Why should you take it? What should you think about if you take it?

Text answers:

This is a lot of work and you need to work as a team to get even close to succeeding.

Students that have a lot of free time and no more than 1 other course that isn't problem solving and programming under pressure could consider taking this course. This course provides hands-on experience in developing a real robot and if someone loves a challenge and is into robotics this is the course to take!

If you are interested in robotics, hands on stuff, want to try out how to deal with a real world problem and have A LOT of time, do it.
If you do not the have time (at least 20h/week and even more when a milestone got close), or you have a lot in parallel, you should maybe not do it.

You can learn a lot and if you are lucky and are placed in a good group you can even have a lot of fun.
But be aware, we had no influence on who is in the group so some of us got lucky and some got just some random guys who didn't give a sh*** about the project course... which then made life really tough.

This course puts its emphasis on coming together as a team and building something that doesn't actually work, but it like totally did yesterday in the lab. It is a stark contrast to most other courses that which tend less practically oriented, and less open to setting your own goals and agendas. It is one of the better courses at KTH.

This course will teach not only about all parts of an autonomous robot but also and most importantly it will make you realize how difficult integration can be.

The course provide a really good chance for everyone to learn programming and robotics. With detailed instruction at first, even people that do not use ROS before can quickly get how ROS works and how to use ROS to build robot project.

The course is the most realistic project that one can get during an education. You have to consider the amount of time and energy (both physical and mental) required to meet the course requirements. But this should not be considered as an "evil" course made difficult just to make so. It roughly shows what is going to be in our professional lives during projects, that is why it is difficult and this is why I recommend potential students to take it.

Integration could ruin everything. Start early.

Anyone who wants a challenge in robotics. If you take it be minded that it is a hard working course. That means spending a lot of nights there working with your teammates. This is important.

Students have knowledge in C++ and ros, C++ at least. Have taken class at least one of applied estimation, computer vision

If you want to put into practice what you've learnt over the last year then you should take this course. I would recommend it to everyone. Try to take easy courses at the same time otherwise you will be in for a world of hurt.

Everyone who is willing to spend a lot of time should take it. You get to work on a project from scratch to finish, meaning that you get an insight into how it is to be working with this after graduation. It is okay to take another course that requires a lot of time in period 1 but not in period 2 since you will want to spend as much time as you can on this project.

This is the best course I took at KTH. It has almost everything related with robots.

This course is for everyone, who likes working in teams on challenging robotic tasks and is willing to spend the time needed for it to be a fun and a successful learning experience. Having good programming skills is the most useful ability you can have. Pre knowledge in some other robotic related topic like estimation, vision, control is very nice to have but it also goes without.

Of course the robotics students should take it as the practical implement of what you have learned about robot. On the other hand, it is a very interesting and challenging course, you definitely can learn a lot!

This course teaches you about real world application in robotics. You should take this course if you like robotics, team efforts and a hard task.
If you take this course, be prepared to sink lot's of hours into it, and start to do it early. Try to be ahead of the milestones because you will need the buffer for completing the product after milestone 4. Also start with the main connecting code base as early as possible. Do no think it is easy, it is not!
Give this course a lot of time and heart and you will get it returned!

The people that should take this course must be willing to spend a lot of time working on the course, never giving up, since there will be many difficulties that you want you to quit...
If you are planning to take it, try to select courses that are "heavy" in the first period, since you would probably not have so much time for them in period 2 anyway.

You should take this course if you are the least interested in how robots work and would like to try to build one. If you think you might have a lot of things to do, besides this course, you should think twice - this course takes time!

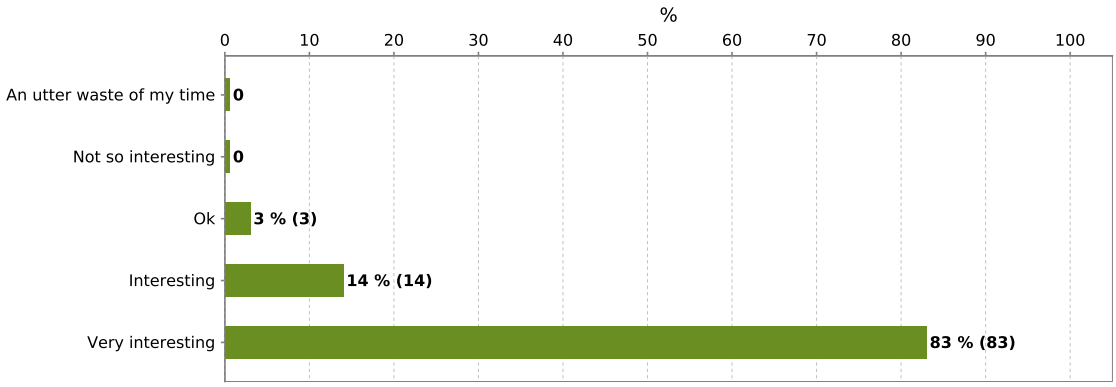
This course is for everyone who has enough time to spear and always wanted to have a hands-on project. But be aware that for three months it will take over your life.

Take this course if you want to deal with a real project to make use of your knowledge and get some hands-on experience. Also, it will give you an understanding of the challenges related to the field of robotics, so if you are thinking of pursuing a career in this field you will get an idea of what you do and do not like about it.
Don't expect to do only what you like: you will have to do what is needed to carry on the project. If you are a computer science, expect to have to code a lot. In any case, expect to spend A LOT of time in the lab.
The project is difficult and full of technical risks. The final product of your work could be not very satisfactory. The intermediate results will often be frustrating. Don't worry about your grade, your effort will be recognized, but you should worry about your motivation: you have to like challenges, not just nice results; dealing with problems, not just finding solutions. If you accept this, and the fact that it will take A LOT of time (have I said this already?), it will be a lot of fun and a priceless lesson of engineering.

Very interesting course, lots of discoveries, but be ready to spend a huge amount of time.

21 have answered of 43 (48%)

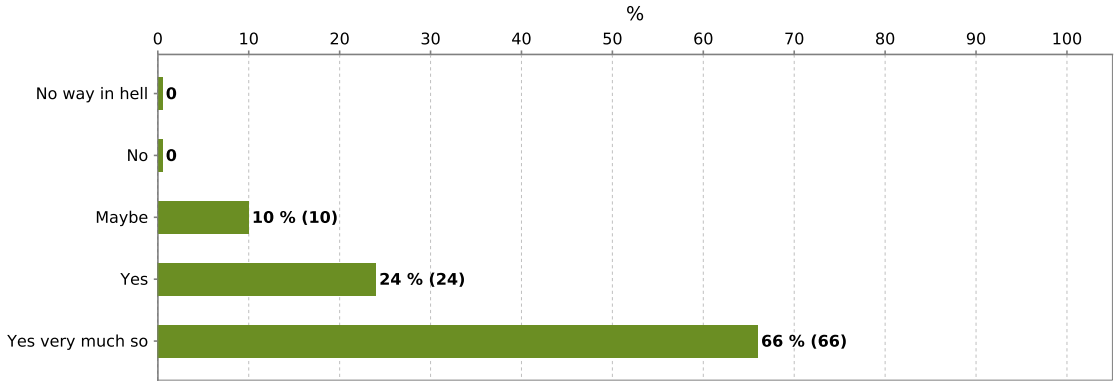
Did you find the course interesting and meaningful?



Answer choice	An utter waste of my time	Not so interesting	Ok	Interesting	Very interesting
Distribution	0%	0%	3,4%	13,8%	82,8%
Number	0	0	1	4	24

29 have answered of 43 (67%)
Maximum number of choices: 1

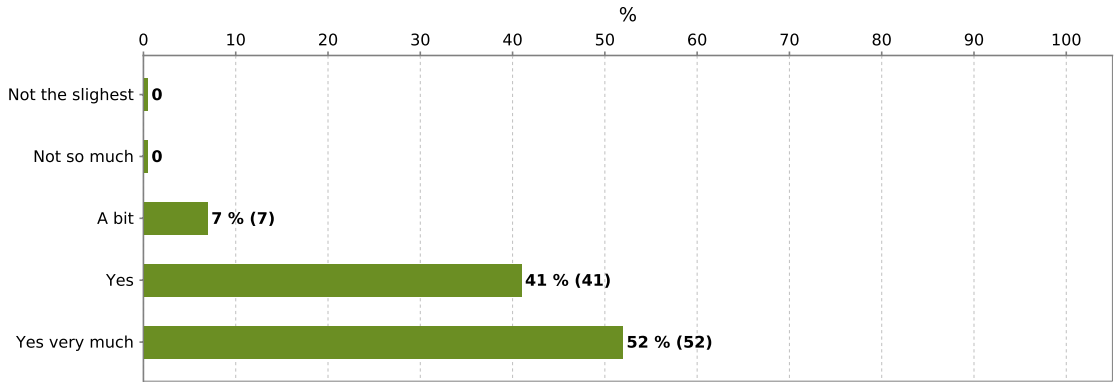
Would you recommend this course to a fellow student?



Answer choice	No way in hell	No	Maybe	Yes	Yes very much so
Distribution	0%	0%	10,3%	24,1%	65,5%
Number	0	0	3	7	19

29 have answered of 43 (67%)
Maximum number of choices: 1

Did this course make you more interested in robotics and robotics research?



Answer choice	Not the slighest	Not so much	A bit	Yes	Yes very much
Distribution	0%	0%	6,9%	41,4%	51,7%
Number	0	0	2	12	15

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

You get a good sense of how hard it is to integrate stuff and things almost never work as expected or intended.

The course

I was already very interested in robotics and this course was actually one of the main reasons I picked "Systems, Control and Robotics" master program. And now after the course, I feel more confident and with more knowledge as well as experience on mobile robotics.

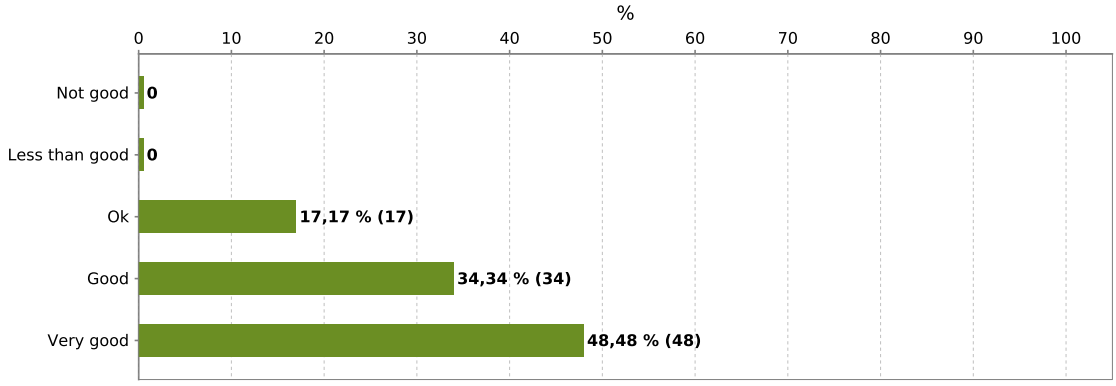
Yes it made me think of all the potential tasks and problems robots face.

This gave a very good introduction to the subject, and gave an overview of all the major areas of it.

In a sense it made me less interested, but in a meaningful way: I realized that I don't want to work with robots full time (or I should say "full stack"), but there are very very interesting applications for artificial intelligence there

---

What do you think about Patric's lectures?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	17,2%	34,5%	48,3%
Number	0	0	5	10	14

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

A little disappointed as the AI lectures where definitely more interesting. The lectures felt sometimes like a lot of topics shuffled together where the structure was lost.

Typical Patric standard lectures.
Joshua's lecture was ok as well. It would have been nice to know in advance that he will talk about the ongoing research and maybe the others TA's could have been there at the end and explain roughly what they are doing... instead of just showing some videos... and when we ask questions he couldn't answer all of them and said we have to ask them ourselves....

Patric comes across as approachable and well-versed in his field. While the course is largely practical, the lectures serve as an important overview of what methods are available.

They were nice and compact. Not too brief nor too in depth, just as needed.

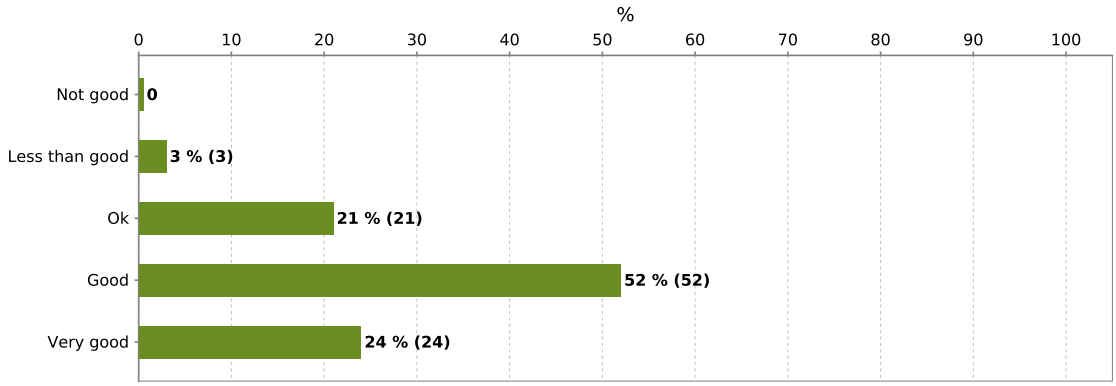
I think I had coincidentally taken courses that covered pretty much all the subject matter in some level of detail already, but they were still a solid introduction to each area of the system.

Very good for the available times. I felt happy to have taken in depth courses in the different robotics areas before. Otherwise, I would have had a hard time understanding the topics. I liked that it was often more practice related, as these issues weren't discussed in the in depth, more theoretical courses.

I found them quite superficial, but it was ok for a project based course

---

What do you think about the connection between the lectures and the project? Could you use what you learned in the lectures in the project?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	3,4%	20,7%	51,7%	24,1%
Number	0	1	6	15	7

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

The lecture provided some understanding of what it was you where using. But as the time was so limited you could not really play with different ideas or risk implementing some newer interesting approach without risking the project to fail.

More information and examples at the beginning would have been nice.
Showing us the UAVs and the other robots was cool, maybe you could spend more time on explainig stuff in them so that we get earlier an idea about things like SLAM or navigation

The freedom of choice is limited by the application, so groups are likely to choose the same methods and to be "inspired" by each other, for better and for worse. There was a clear connection between lectures and project.

I guess only using the knowledge from the lectures would not be enough, since the lectures covered the subjects from a higher conceptual level. This is how I think it should be, because otherwise it was not possible to cover everything in the lectures. We have complete separate courses to cover some of the subsystems such as Applied Estimation, Computer Vision and so on. Yet the lectures guided us to look where or consider what in implementation and I think that was very useful.

The lectures are a good starting point but you will have to do your own research into which algorithm to use (or ideally you will have some experience from another course).

I think it's good to get a broad overview on the robotic topics in the lectures such that everyone has an understanding on what the different parts are about. I feel like the connection between lectures and project is hard to make. In some parts the lectures might have given some hints but the methods used were mainly researched for in the internet.

You could certainly use the information in the lecture in the project, but biggest problem was that the lectures was in the beginning. This meant when you actually needed the information it was a long time since the lecture. Also, the lecture should maybe focus in the actual hardware used in the project when discussing the topic. Example: when discussing range sensors maybe focus more on the LIDAR than other sensors not used.

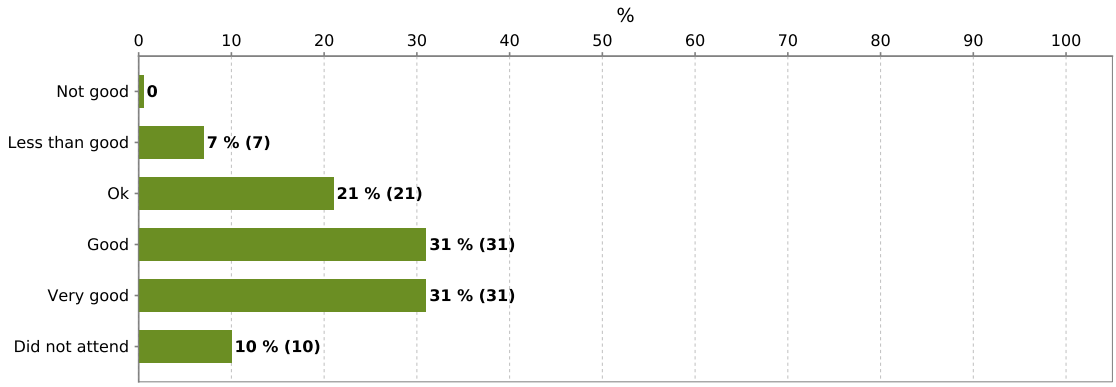
Sometimes a bit more evaluation of different methods would be useful (as done for Particle Filter and Kalman Filter). For instance, when are certain methods applicable or what needs to be considered when using them.

Yes, but lectures material was very simple and we had to look up a lot. I think it would be good to learn git for the course.

Although not very detailed, the lectures definitively gave the right direction

---

What do you think about the tutorials with introduction to ROS and the programming lab (LAB1)?



Answer choice	Not good	Less than good	Ok	Good	Very good	Did not attend
Distribution	0%	6,9%	20,7%	31%	31%	10,3%
Number	0	2	6	9	9	3

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Tutorial was held in a satisfying manner but LAB1 itself felt useless as it not really connected to the project and only was to learn some ROS (Which is good).
A better Lab 1 would be to set up some simulation of the project which can be later used to develop in parallel.

This was way too late.

It was useful. I'm a software engineer, so it likely wouldn't have been a problem to

I already knew ROS hence personally, first lab was very trivial for me. Yet if I was seeing ROS for the first time, than I might have some difficult times to understand the concepts, go through online tutorials, write nodes and present them in that short time. Because I remember in my bachelor's it took me some time to even understand wtf is ROS, is it a program, or a language or what I am supposed to do with the things in online tutorials and so on. From that perspective I think it might be better to provide a document with explaining ROS and its functionalities in a robot system in a way that ros tutorials or wiki does not provide. Because I saw some people were still very confused about ROS even after weeks, complaining that they don't know ROS or understand it. And it is very annoying that when they don't have the grasp on it, they cannot contribute to the project work, resulting in tough situations and anxiety. It is not that hard to pass lab 1 even without a full understanding of the tools of ROS. Many students are used to fill the function code blocks with given definitions of inputs and outputs. It works at the end yet their grasp on the dynamics stay limited. When it comes to the project, then you have to think what to use, what should be the input to the node, in which message type, should it be a topic or a service or actionlib... Then people got stuck and can't contribute.

I thought it would have been better to have one or two lab sessions, there were some things that outright weren't stated in the tutorial and a couple of things that took me quite a long time to find online. I feel this could have been solved very quickly in a lab session, also the collaborative nature would have allowed some of these issues to be resolved very quickly.

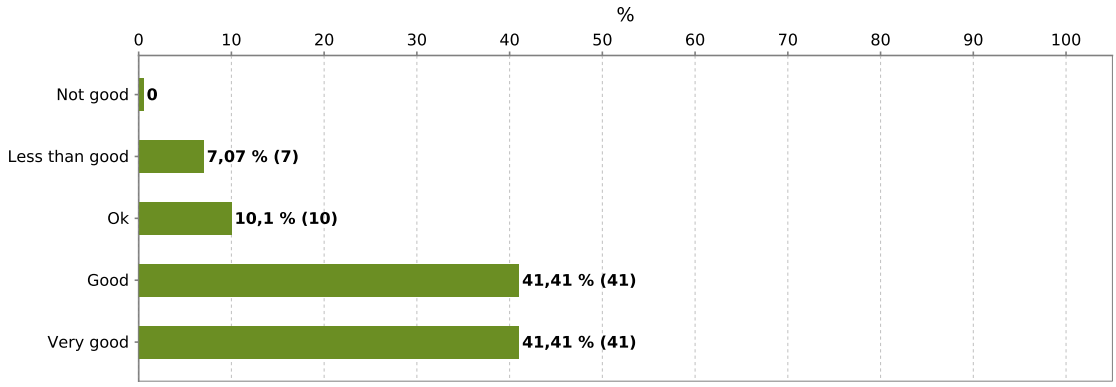
They were a very good starting point for learning ROS. I would consider if the lab would give a better set up for simulating the real robot in the end.

When I have attended them I had already started with the lab, so the tutorial didn't give that much. However, it was nothing wrong with them, and I recommend keeping them to the next year.

I wouldn't call them tutorials, just some hints were given, but they are not so meaningful when you still have to learn the technology. Anyway, I don't think this is a problem (see next answer)

The lab was very good, however the time between the end of the lab and the start of the project was a waste of time. The project should have started much sooner.

What do you think of the ROS programming assignment? Did it help you get started?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	6,9%	10,3%	41,4%	41,4%
Number	0	2	3	12	12

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Felt more like a chore and the usefulness for the project was not there. Good to force people to do ROS but this way is suboptimal.

I felt like doing nothing the first few weeks, then there was this lab.... then nothing and then it went crazy.
It would be better to have scheduled the lab much ealier and maybe split into two parts so that the students have to do some small part on their own... then some follow up explanation on ros and then more lab stuff

It was very useful, and gave us common ground when starting the project.

Same as my previous answer.

Should be more assignments

Yes it was quite good, it helped me learn the ROS basics.

Then it's assured everyone knows the framework.

They were a very good starting point for learning ROS. I would consider if the lab would give a better set up for simulating the real robot in the end.

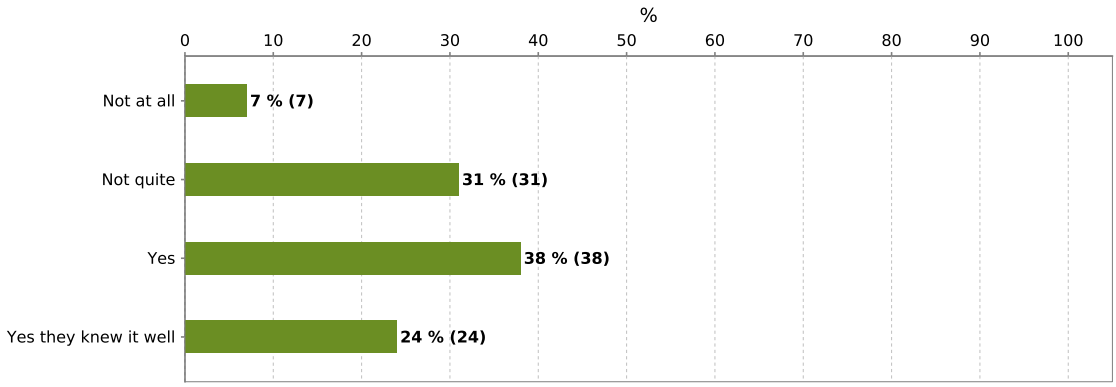
Maybe give an more explicit hint to look at the implementation provided for creating new nodes and code since the provided code is well structured and uses classes. Through the tutorials one is likely to get automatically generated code without understanding its purpose.

Yeah if you mean the Ros tutorials they are good.

I think this was the perfect way to learn the technology: lectures about this kind of stuff are usually boring and useless

---

Do you think that people in the group knew enough about ROS when the project started?



Answer choice	Not at all	Not quite	Yes	Yes they knew it well
Distribution	6,9%	31%	37,9%	24,1%
Number	2	9	11	7

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Some knew way more than others.

Not all of them.

None of us did, but that's the point of education. It was enough to get started. I would have liked it if people had been trained in using Git beforehand.

It was me and another team member who had ROS experience. Other two did not have any and I suspect one of them still might not have enough to contribute to any such project.

Some of they very well and the others completely not

I think that the tutorial covered most of the things we needed to know. There were only a couple of other things we needed in the actual course like how to use rosparameters and make launch files but this could be found and understood extremely easily. The bread and butter of ROS was taught in the tutorial.

Definitely enough, learning while doing is nicer.

The basic knowledge from lab1 was there :)

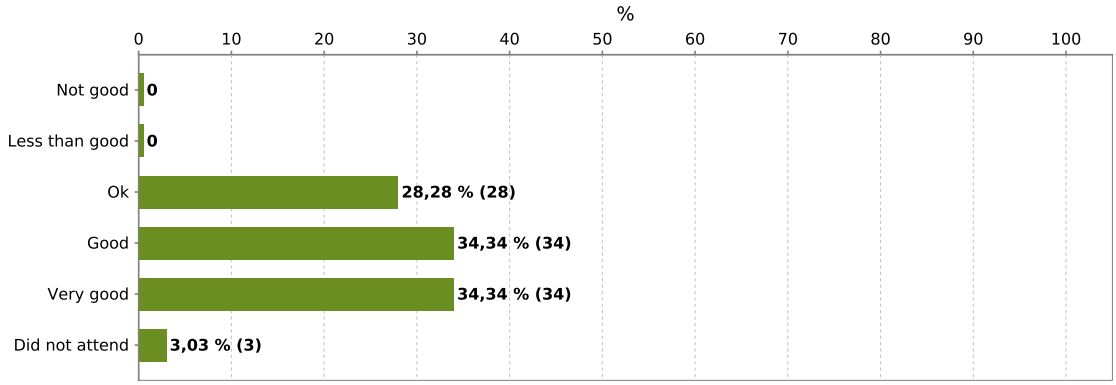
Maybe have some sub-assignment about writing more "object-oriented" ROS. By this I mean having the node as a class. It required some information to get it right, which I didn't have when I did the lab assignment.
The primary reason for this I think it made the node classes which simpler, cleaner, etc. compared to a more procedural way.

Enough to get started and the rest could be learned later or through others.

No me included we didn't know much about it.

Some did, some did not. There is a problem here, but I am not sure what it is.

What do you think about the first introductory session in the robot lab?



Answer choice	Not good	Less than good	Ok	Good	Very good	Did not attend
Distribution	0%	0%	27,6%	34,5%	34,5%	3,4%
Number	0	0	8	10	10	1

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Felt chaotic as a lot of stuff still needed to be set up and not everything was working as expected.

It was very straightforward, we have already made the connections, ran the motors and read from the sensors by the end of the session. Liked it.

Yes this along with the documentation online was perfectly enough to understand all parts of the system.

Brief but sufficient

I would not say it was essential but it was nice to be showed how every component worked!

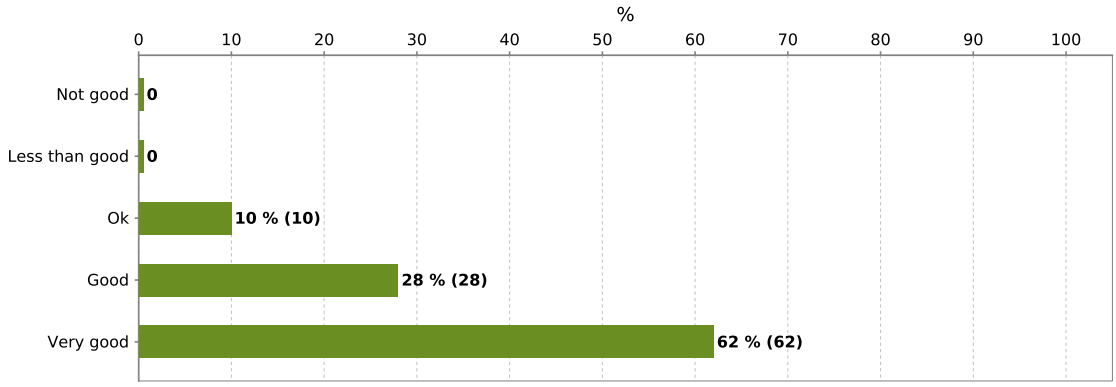
The necessary information was given to start building the hardware of the robot.

But the slides to that lab were provided very late and most of the information could not be found on the wiki.

Don't remember

The minimum required

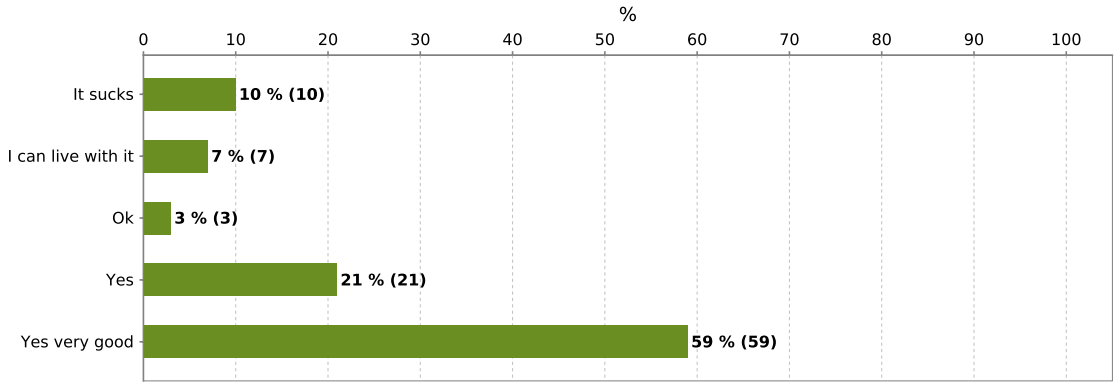
How did you like the project?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	10,3%	27,6%	62,1%
Number	0	0	3	8	18

29 have answered of 43 (67%)
Maximum number of choices: 1

You were divided into groups partly based on your skills and NOT knowing one another before. Do you think that this is a good way to do it?



Answer choice	It sucks	I can live with it	Ok	Yes	Yes very good
Distribution	10,3%	6,9%	3,4%	20,7%	58,6%
Number	3	2	1	6	17

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

This is the biggest flaw in this this course. I get the reasoning that you will have that later in real life and that it is good training and so on. But the project is so demanding and intense that you need a working team that pulls in the same direction. And while it is important to have different skills it is far more important to have a working team to split the work or help each other. Not knowing the other team members and there working pattern can be absolute destructive. People can have completely different priority and are forced to deal with it. There is also no system in place to really detect a not working group as the teamwork is such a big part of the grade and the milestone where super hard.
The fear for letting the students choose there own group resulting in A and B groups makes sens but doesn't really matter as the resulting teams given the current system resultet in A and B teams ... just that it was clear that it was not the students fault as they had no control over the team selection.

Not knowing one another is more important than the skill based division. Otherwise, subgroups are created within the group.

This was like a pre-decision on the final grade made by the teachers
you argued that in real life you cannot chose your colleagues... but there you can do something if they do not work as they should...

I honestly could not think what to select in the answers of this question. The idea of the test for skills and separating the groups based on it sounds very reasonable. Yet i really doubt it works. The whole point is to balance the teams so that there would be at least one person on each sub-system that knows what is going on and shuffle the groups so that people would have to work with strangers so that it would be more realistic. It is very good on theory. Yet the groups were not balanced at all. First of all there were groups of five (or even six if I remember right) people. And it is hard to actually evaluate one's grasp on topics just from the test. The test itself was fun to make, yet I think it was irrelevant of what we did at some points. For example, there were questions on bit masking, or unnecessary pointer operations. I guess the test

was outdated from times where the course was based on programming microcontrollers. The questions on the test can be updated including ROS, object oriented programming, or more qualitative questions on Linux such as commands instead of asking what is the most recent kernel version. I believe the separation based on skills is a good way, yet the test should be changed in my opinion. But I'm sure that, at the end it is more or less luck to be in a team that is effectively working or otherwise.

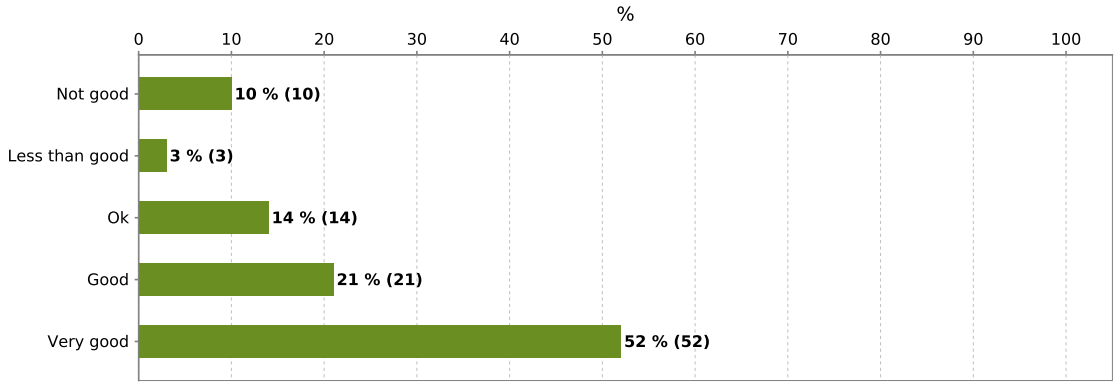
Yes absolutely. We had a person who had an expertise in each of the main areas of the course, this worked well.

This way you get to know new people and especially the many exchange students have the chance to get to know the locals. However, I could also observe, in other groups, how the groups composition seemed to influence the members participation and motivation.

Maybe add a question with what grade someone is aiming for.

It is so random depending on how the group ends up working

How did your group work together?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	10,3%	3,4%	13,8%	20,7%	51,7%
Number	3	1	4	6	15

29 have answered of 43 (67%)
Maximum number of choices: 1

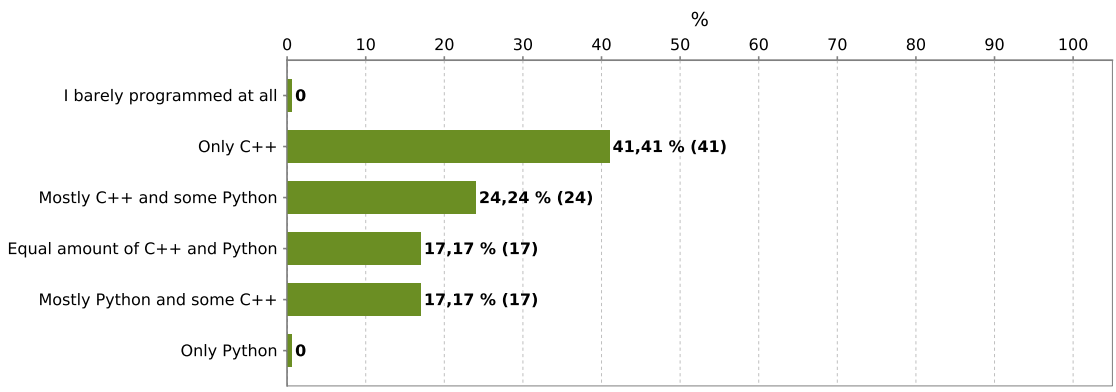
Respondents comments:

Fantastic group every one was eager to work and on the same page.

We worked well. Probably should have had regular meetings or something

It was often like hell. But I learned and improved myself a lot from that situation. I liked all my team mates quite a lot in person, we had fun times when we were not talking or working on the project. Yet for some reason we had very bad team dynamics. Sometimes, some people just can't work together no matter how much they try. We were constantly on edge of failure. This happened because of the different expectations on the course and project. The work load was very uneven and responsibility levels or understandings were disparate. Yet we sticked together, we tried our best. We tried to overcome the problems, but sometimes it does not work. Luckily the TA's and Patric saw our efforts on that.

What did you program in?



Answer choice	I barely programmed at all	Only C++	Mostly C++ and some Python	Equal amount of C++ and Python	Mostly Python and some C++	Only Python

Distribution	0%	41,4%	24,1%	17,2%	17,2%	0%
Number	0	12	7	5	5	0

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

used the chance to learn python -> made it harder as the time was so limited.

Hey, I'm a Python guy.

Programming in different languages may provide difficulties in integration.

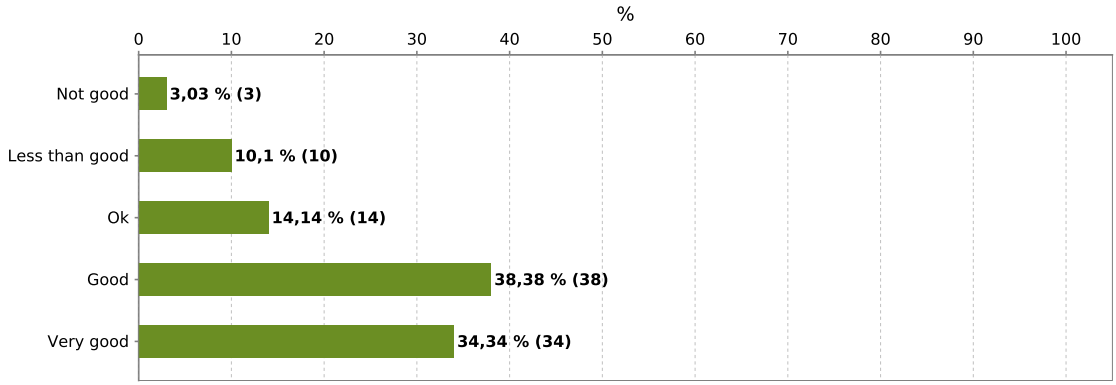
Our code was completely in C++ because of the personal choice.

I tried to avoid c++ when possible (because I don't have much experience and find it difficult and very time consuming) but this was essential for the PCL (point cloud library). I did the rest in python. Anything computationally intensive will probably have to be done in c++.

Python is a lot faster for debugging. Some things, like action lib only seems to work in c++

We tried some python, but switched back to c++ quite fast.

What did you think about ROS?



Answer choice	**Not good**	**Less than good**	**Ok**	**Good**	**Very good**
Distribution	3,4%	10,3%	13,8%	37,9%	34,5%
Number	1	3	4	11	10

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Helped a lot but also the source of many many many frustrating errors.

Th

ROS was ok but only after you knew the traps...and if you knew a guy who could help if weird things happened

ROS is a sad joke in many regards. It helps somewhat, in that it gives a framework to start with, but it has the weirdest bugs and integrates rather poorly with the rest of the operating system.

Very useful tool. Saves from reinventing the steel.

The whole make file thing can be very confusing and irritating. Especially as from what I can tell it looks like you have to recompile everything every time.

I like that we use a system used a lot in the"real" world. I would love for it to be on mac or windows, but that is nitpicking.

Architectural wise, ROS add maybe to much overhead and less intuitive behavior with its callback nature, but overall it's a good framework. It has a lot "in the box" with visualization, recording/playback, libraries etc.

Very cumbersome when you are used to python and python libraries. Hard to find API docs, you always end up in tutorials. API comments not complete. Not reliable and behaving as you expect. C++ and python API not the same naming and not the same functionality. RViz is kind of nice, but software getting random segfaults at startup does not feel very professional.

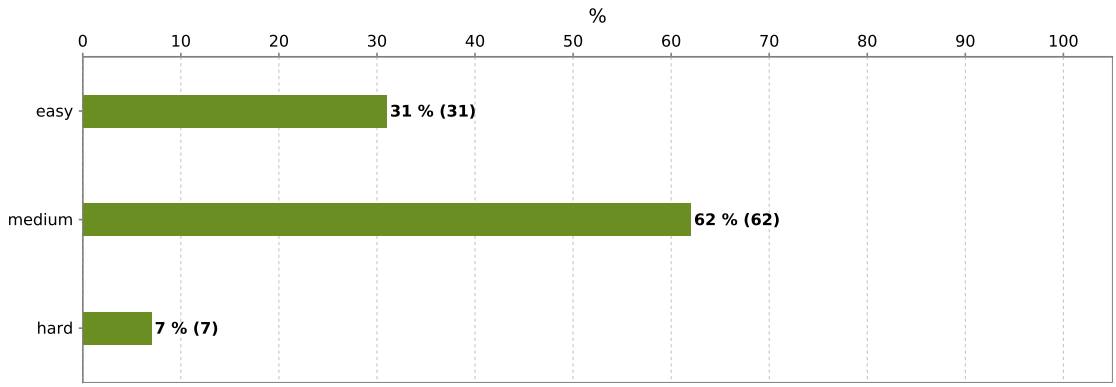
Good documentation and tutorials available.

It is instable crashes alot and somethings have very poor documentation like tf2.

It's messy, with a lot of special cases, inconsisten design choices and I don't think it gives an adeguate level of abstraction

---

How hard was Milestone 1?



Answer choice	easy	medium	hard
Distribution	31%	62,1%	6,9%
Number	9	18	2

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Major problem was not knowing all the rules for the competition and still having to build the entire robot.

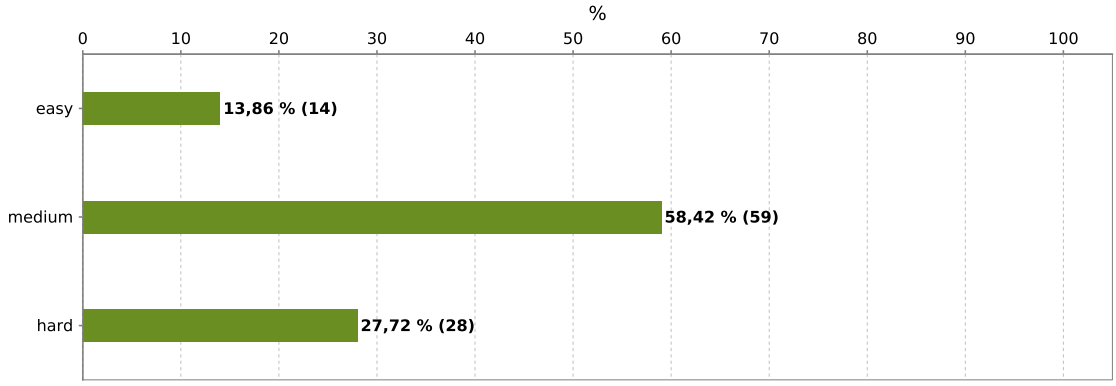
It was the only easy Milestone in my opinion. Pretty straightforward.

Very reasonable.

Could have been harder.

The focus was primarily on building the hardware. The software part was very easy.

How hard was Milestone 2?



Answer choice	easy	medium	hard
Distribution	13,8%	58,6%	27,6%
Number	4	17	8

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

The only reasonable milestone while the part with the map was useless as we redid it anyway.

There was a huge step up after MS1. The problem with the camera as well as the arm was annoying.

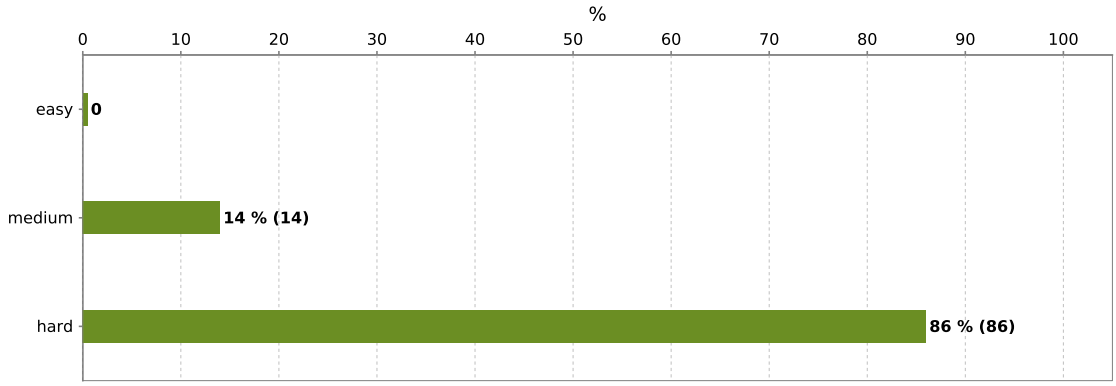
Very reasonable.

The mapping part with saving every Nth scan was a bit unclear and I believe we later scrapped that implementation later in the project...

Content-wise, I felt that this milestone should have more. Since we were 4 people in a group, but the work was more for like 2, max 3 people. Adding something from milestone 3, or something with motion such as following simple waypoints would be good.

However, the original version (with being able to pick up an object) would have been very hard.

How hard was Milestone 3?



Answer choice	easy	medium	hard
Distribution	0%	13,8%	86,2%
Number	0	4	25

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Absolute impossible. This was really insane no one could do that in time -> the grading was then way to lean to counter that fact and resulted in extreme confusion.

MS3 was basically everything required from the project. Hence it was hard to catch up with.

I sort of ended up achievement hacking this milestone because I didn't yet have my vision system working so I purposefully created a very crude method to differentiate between a few of the objects.

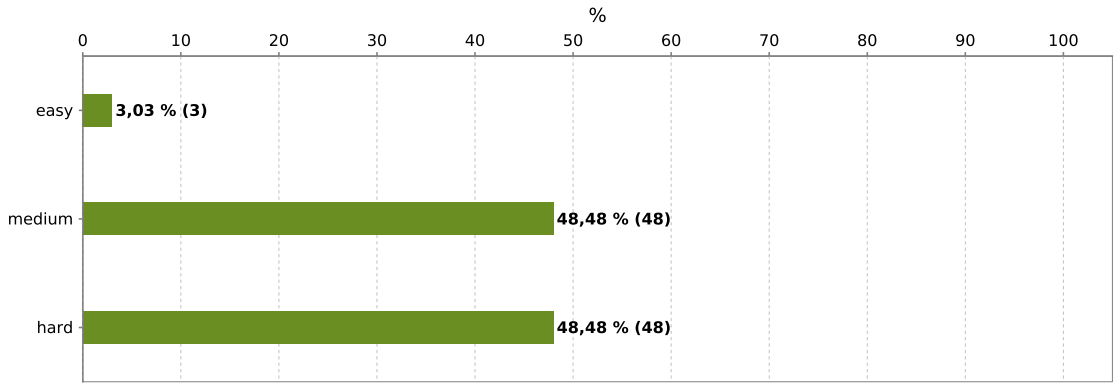
An intermediate step was missing before integration was asked for.

This milestone was placed on the exam break and made the team not put as much time as we should into it...

This milestone had to much stuff in it. Some parts should be moved to milestone 4, or giving more time for this milestone.

It has a lot of different components. And since it was scheduled one week after the examination period we had effectively one week to implement most of it.

How hard was Milestone 4?



Answer choice	easy	medium	hard

Distribution	3,4%	48,3%	48,3%
Number	1	14	14

29 have answered of 43 (67%)
Maximum number of choices: 3

Respondents comments:

pretty hard but not impossible if the team worked well together and stayed away from quick and dirty solution the other milestone incentive to. That there was no camera part was very strange as it made the distribution of work very uneven for this milestone and only working team could counter that.

It was very similar to milestone 3.

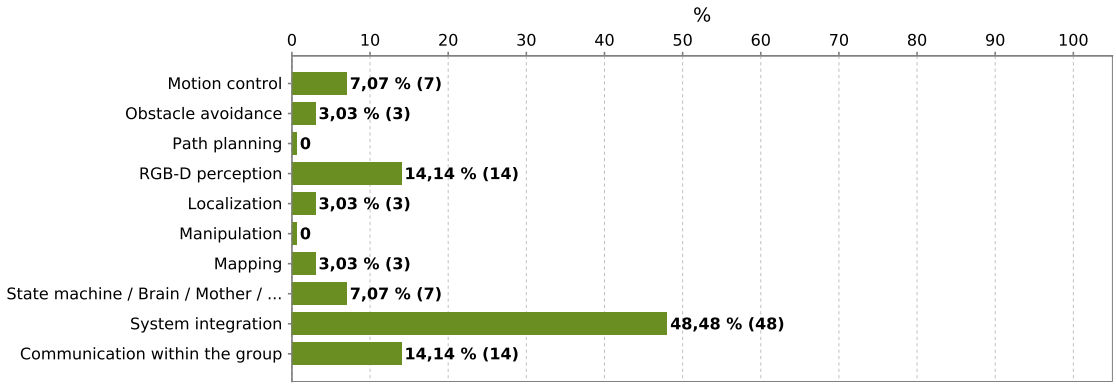
Well, it was meant to be easy as long as we have MS3 completed. That was not the case, we tried to complete MS3 long after its deadline. But for example global localization in MS4 was done in MS3.

Yes this was difficult, this required a lot of integration!

The overall milestone was maybe not that hard, but you worked mostly getting the stuff from milestone 3 to work good.

Provided that the previous milestone had been met. But since it doesn't really reflect what needs to be done to complete the whole task this milestone seemed to be of less importance.

Which of the following parts of the work do you perceive as being the most difficult now that you have completed the project?



Answer choice	Motion control	Obstacle avoidance	Path planning	RGB-D perception	Localization	Manipulation	Mapping	State machine / Brain / Mother / ...	System integration	Communication within the group
Distribution	6,9%	3,4%	0%	13,8%	3,4%	0%	3,4%	6,9%	48,3%	13,8%
Number	2	1	0	4	1	0	1	2	14	4

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

The integration of all systems together was by far the toughest part.

If I would have a second pick that would be communication within the group. Although this doesn't mean that the rest was not difficult. Localization and RGB-D perception was difficult as well but some students had an implementation from taking courses in vision and applied estimation that helped a lot.

Three things that work in isolation is not the same as three things that you can put together as one. I will refer you to Hofstadter's Law: It always takes longer than you expect, even when you take Hofstadter's Law into account.

Very hard to give only one answer. But for me, it was not the hardest to work on localization for weeks, neither spending nights at the lab for the brain, not even the system integration. It was to work with people that exhausted me and took half of my energy. Because you understand that you should have the team integration prior to system integration. One does not work without the other.

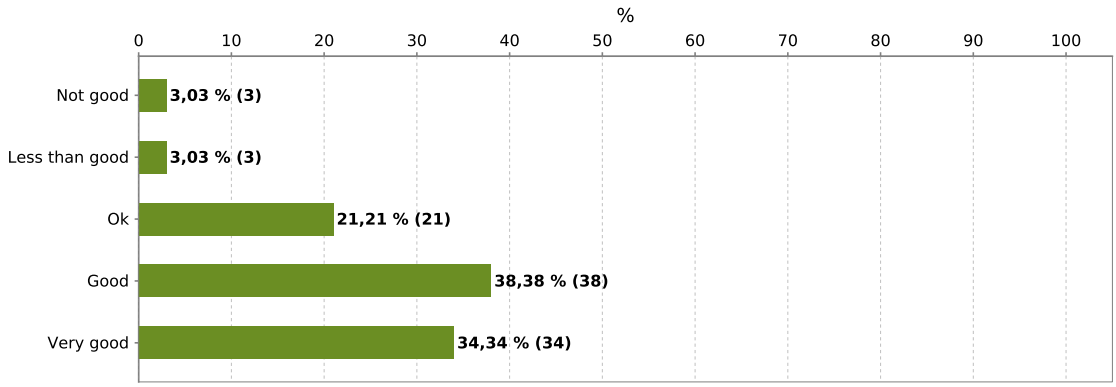
I would argue that either vision or integration is the hardest. Vision because there are multiple algorithms and small tweaks you will have to use together for the 1) detection of objects and locating them and 2) their classification, its almost like a two step process. Also vision is extremely noisy and there is no end to the situations that will trick your vision system. But I may be biased because I did vision. Integration is difficult as each part of the system has its own problems and then combining them together compounds these problems. Also it forces you to think about the higher level decision making processes of the robot which are difficult.

everything can be working separately and then it suddenly crashes when its integrated

The vision system was the hardest to get robust, meaning working in different lighting condition and positions of the robot.
I also felt that motion control / obstacle avoidance was difficult, but you could solve it with a very simple solution, that maybe didn't have such a nice path / took long time to complete the trajectory, but worked well. This is the reason I didn't set this as the most difficult.

Very close to vision.

How did you like the contest event at the end of the project?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	3,4%	3,4%	20,7%	37,9%	34,5%
Number	1	1	6	11	10

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Disappointing result and strange rule set (there should be a way to get points without solving the entire task to avoid people trying all the things and failing while they could have easily demoed parts of the system.

The results were laughable, but it was a really important motivator. I told Diogo this, but I think there should be a two-day competition -- first a time trial where you get to test your wings, and then the next day, an actual competition with booze and pizza like there was this year.

I cannot say I enjoyed the contest. It was very stressful two days and nights before that, everybody in the team was sleepless, tired, angry simply at the edge. We had quite a tension between us. I was sad because I was expecting thousand times better than what we had.

Yes it was fun to watch all the robots!

It would have been very good if it went better for all the groups.

A pre contest would have been nice. Or at least to have some built up test area, showing how important it is to test before in the new environment.

I do not know why so few of us manage to get our robots rolling... IT would have been better if most of us got out of the first obstacle ^^ (More the students fault?)

Very well organized. The maze and lighting conditions was very different from the lab maze, but this is something that you should expect, but maybe didn't think to much about.

It was good, but I think you tried to do it a bit too difficult.

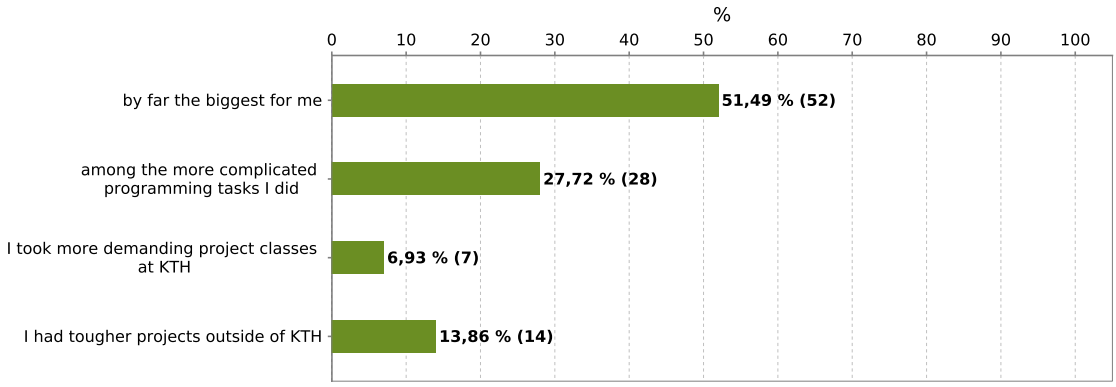
It is nice to have something to work towards to. However, the actual event was slightly disappointing since most robots could barely drive.

Good but would be good with one test run or something

I understand the contest in not related with the grade and is supposed to be very difficult, but seeing how unlikely you are to fulfill the task can be quite bad for motivation

---

How big of a software project was programming the robot for you?



Answer choice	by far the biggest for me	among the more complicated programming tasks I did	I took more demanding project classes at KTH	I had tougher projects outside of KTH
Distribution	51,7%	27,6%	6,9%	13,8%
Number	15	8	2	4

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

In actuality, it is rather small. The components require a lot of tweaking, but in terms of SWE it's not that big.

I didnt find it hard, but time consuming.

Yes This is probably the hardest programming task I've had. But then again this is to be expected as you are programming multiple algorithms and additional code to make all these results interact with each other, over a few months.

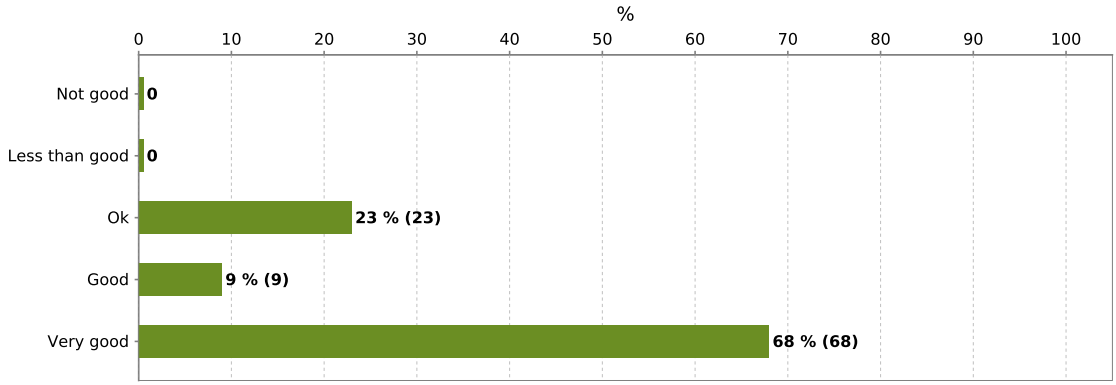
In total it was the biggest code base I have been developing from scratch. But I have worked with bigger projects but then my work was in a much smaller part of it. I have also done more coding my self on other projects, but they were with a smaller team.

Definitely the most demanding and complicated task in any course at KTH. Each individual part might not been super hard, but the overall system did amount to a very complicated system, and adding the hardware to it makes it very complicated!

The challenges are different compared to a pure software project, but still big

---

Groups 1-3: How would you rate Diogo in his role as a TA?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	22,7%	9,1%	68,2%
Number	0	0	5	2	15

22 have answered of 43 (51%)
Maximum number of choices: 1

Respondents comments:

The 2 week where he was gone was very unfortunate as it was close to milestone 3.

It felt sometimes as he did not listen and only heard what he wanted to hear despite of what was actually said.

He was approachable, honest, encouraging, and upbeat.

I wish there was an option for "excellent". I really liked Diogo, he was very patient for us, explaining and helping us at times we need the most. He is very talented and kind. Sometimes I was amazed how he was handling that many things so calmly.

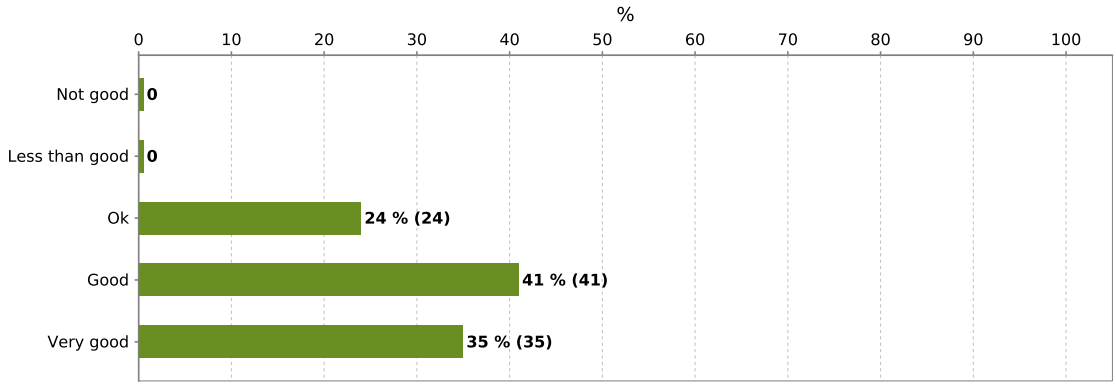
Yes he seemed very knowledgeable and had lots of practical advice.

Did not have much interaction with him :) He seems good.

Accidentally clicked on this button, and can't unselect it :(

I was so focused on the work

Groups 4-6: How would you rate Silvia in her role as a TA?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	23,5%	41,2%	35,3%
Number	0	0	4	7	6

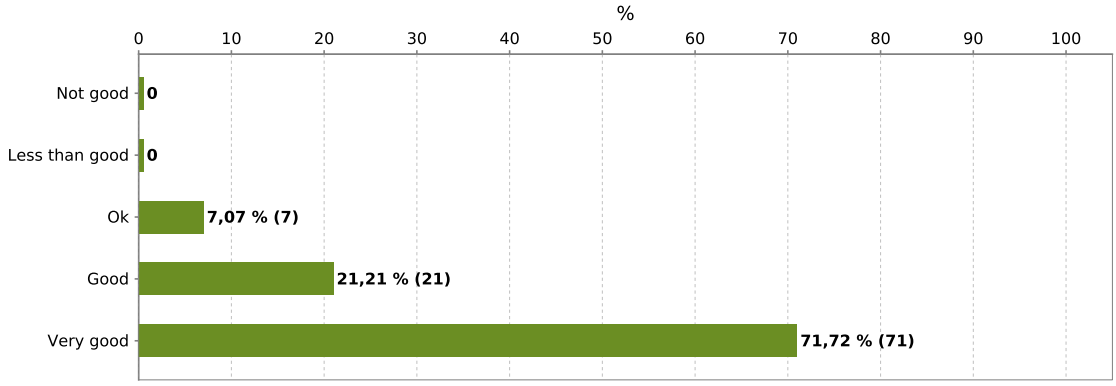
17 have answered of 43 (39%)
Maximum number of choices: 1

Respondents comments:

N/A

Did not have much interaction with her :) She seems good.

Groups 7-9: How would you rate Joshua in his role as a TA?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	7,1%	21,4%	71,4%
Number	0	0	1	3	10

14 have answered of 43 (32%)
Maximum number of choices: 1

Respondents comments:

N/A

He were very responsive on email, and I believe my group got all the help we needed.

Comments in general about the robot hardware (NUC, motors, sensors, arm, ...) and how it could be improved

Text answers:

The arm is really horrible. The new package by Joshua helped but came a little late. Also it should be mentioned that one must think about task computing and how to divide the cpu cycles in an effective manner.

I didn't use the arm but I heard it was pretty terrible. Some advice for people using the camera: immediately lower the framerate as described on the course page, if your driver is working then never update your version of ubuntu (this caused mine to crash), also the version of ROS we used as well as the PCL uses c++98 so don't use c++11 language features and then change the makefile, because any PCL function will give you an immediate segmentation fault. This isn't an suggestion for improvement, just a warning to future students using the Realsense sr300 camera.

I think the equipment was very good and I don't want to sound ungrateful but a better arm would be nice.

the uarm was anoying

There is probably not one part on the robot that didn't eventually fail in some way. Contrary to what many might say here, the uArm control was not that bad once Joshua's wrapper was in place. We could pick up the spheres reliably.

The wheel hubs are terrible, the arm was both buggy and had bad power connection. I loved the NUC. We also had some problems with the motor controllers and lidar dying (propably software problem)

The sensors are great. The connection between wheels and motors are not very stable, especially when robot is moving.

NUCs are great, was expecting way more primitive boards. Generally, the boards, drivers, motors, rplidar are all good. Camera was having problems so we had to reduce the fps rate. Arm was shitty and not reliable at all. Motor hubs to connect the wheel and the metal thingy on it were very annoying, it was coming out all the time resulting in robot to lose a wheel on the way.

It is good as it is. Maybe better motors and wheels, as well as better arm could be good improvements.

improve the arm core

There were a lot of different problems with most of the hardware but in a way I liked it that way since you learned to figure out how to solve the problems which you probably will have to do later in life if you work with this.

The arm is not the best one but everything else works pretty well.

Motors: Phidgest ROS wrapper often dies.
NUC: Faster might be an advantage
Arm: Change of arduino software would be nice, such that it does not move randomly at power up.

I like the level of development that was done before to allow the project to just be in the ROS-layer.

Arm: The latest revision of the arm controller worked ok, but was still issues. If the arm will be used next year, you can maybe write a new controller before the course starts.
Other than that, the hardware worked great.

We had some problems with motors that crashed and that the camera stopped working after ~1 min in the round. I'm not sure if it was hardware related, though. We also had problems getting the arm to move down deep enough, but that mainly because it was mounted with an angle, I think.

Motors died very often on startup and needed to be restarted. Cannot rely on arm servo encoders and do small corrections.

There was not a correct specification of the motors (and their encoders) available which required extra tuning to find the right parameters. Motors with a more reliable specification would be nice. The nodes for the motors also frequently died either during start up or during driving.

The hardware is fine I think, but the firmware could be improved

Hardware was mostly good.
The wheels and the whero hubs could be improved.
The arm had many troubles for us, but that does not necessarily mean that all such arm are bad.

---

20 have answered of 43 (46%)

Comments in general about the project

Text answers:

The task and the topic where very good but the milestone and set up where bad and didn't contribute to solving the task as it made it nearly impossible to be creative or think of non obvious solution. The presser of the milestone and the set up that if you of pas at that date to receive a penalty made things even worse as it generates tension in the groups if the milestone does not get passed because of one member.

Maybe the lectures should be in parallel with the milestones and include more detailed information.

- The project was cool but it was hard to stay focused on the interesting part of the project when you had a guy in the group that just talked, screwed things up claimed credit for work he didn't do...
- you should not change too much from one year to another
- maybe check that peolple do not take too many courses in parallel
- the exam was anoying and just stole valuable time that could have used to study for more important exams...
- emphasize what a STATE MACHINE is, give more examples

The lab environment is unacceptably loud, crowded and the "air" quality is that of breathing into a plastic bag.

The project provides excellent hardware and instructions for me to learn ROS, robotics, and programming.

change the exam time maybe

See all the other questions.

A lot of fun.

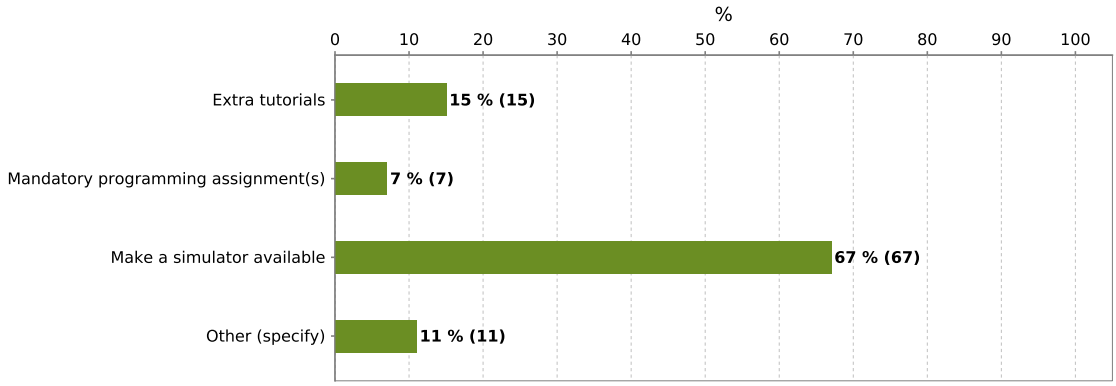
It's really demanding but also really enjoyable when its over

It was both fun and rewarding.

Very good, challenging and demanding. It was a bit chaotic in the robot lab because it was too small, but the mood in the room was always very good. Everyone were very helpful and there were people there almost 24/7

12 have answered of 43 (27%)

Which of the following ways would be the preferred way to improve your ability to contribute to the programming of the robot?



Answer choice	Extra tutorials	Mandatory programming assignment(s)	Make a simulator available	Other (specify)
Distribution	14,8%	7,4%	66,7%	11,1%
Number	4	2	18	3

27 have answered of 43 (62%)
Maximum number of choices: 1

Respondents comments:

The difficult part in the project was understanding ROS in case we didn't have any prior experience. I think that tutorials regarding ROS should be made available to the students. Having a solid understanding of ROS would lift a huge weight from the programming that the project required.

I actually did make a simulator, it helped the group work in parallel.

There was no way to improve my ability to contribute to the programming of the robot, I guess I programmed 80% of it. Yet as I mentioned in the answer for lab1 question,I think it might be better to provide a document with explaining ROS and its functionalities in a robot system in a way that ros tutorials or wiki does not provide.

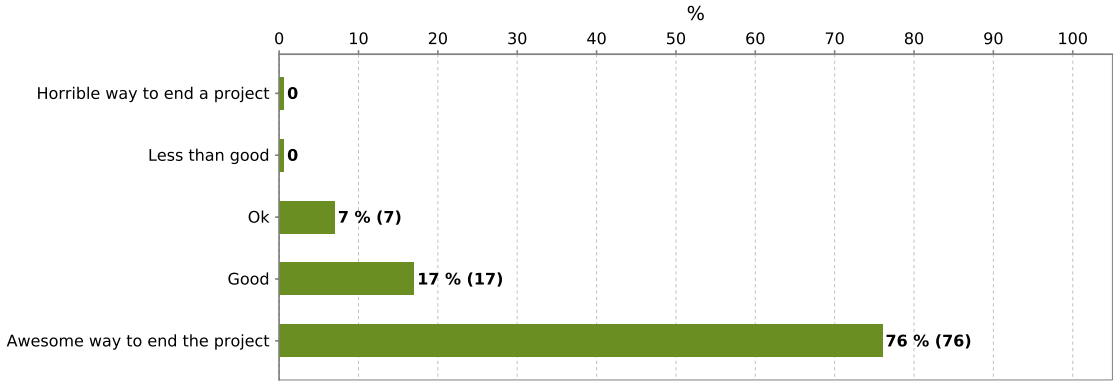
Hmm, I think if you have some programming knowledge (which you should from other courses) and have completed the ROS tutorial, then the rest is really just up to you to get on with it.

Especially in the end, a lot of time was lost due to having to coordinate who is to work on the robot.

Overview on available libraries with suggestions on what has good chances to produce results

---

How did you like having a contest at the end of the project (try to disregard how you did in the contest)?



Answer choice	Horrible way to end a project	Less than good	Ok	Good	Awesome way to end the project
Distribution	0%	0%	6,9%	17,2%	75,9%
Number	0	0	2	5	22

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

The pizza was an awesome touch after spending the previous night at the lab.

you already asked that!
It was nice and the pizza was good

The way that contest is organized is very cool, it is stated in the beginning of the course that it is only meant for fun and not for grading. I think it is very good and wish we did better.

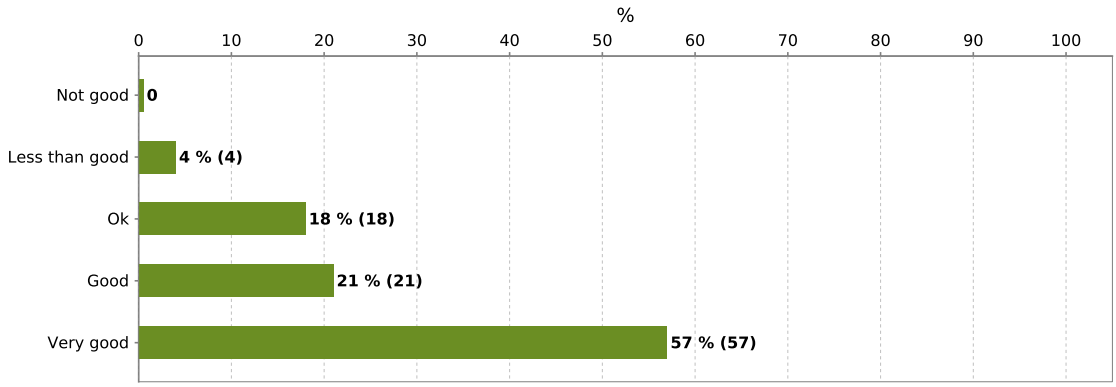
Even though our robot failed miserably in the competition it was clear to me that the aim of the course wasn't to have a perfect robot that aces the task. The competition is a fun way to see everyones robot in action (and get free pizza!!!)

Really gives the extra motivation. Of course you spend some time on things which only serve trying to win the contest. But it also brings together the whole team and also class as everyone is working on it so intensively. An idea could be to have an additional small half way contest to achieve these things even earlier.

Having a clear deadline makes you plan better in my opinion. It also gives you some relief when the contest is over knowing that the course is approx. over! I also thought it gave a good "team spirit" working the last days before the contest to get the robot ready.

It was a very clear goal point in the project, and the result of it really shown how difficult integration is.

What did you think of the criteria based grading system?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	3,6%	17,9%	21,4%	57,1%
Number	0	1	5	6	16

28 have answered of 43 (65%)
Maximum number of choices: 1

Respondents comments:

Should be clearer stated what is to be expected as well as a way to counter a bad team should be implemented as it gets impossible to get a high grade without having all the milestones completed.

Would be ok if you would stick to what you said/wrote

I like that I knew from the beginning that what matters and what should I emphasize on to improve myself. Very structured grading system.

Yes it seems very good, everyone has to pass the first part of the course, this should give them a basic understanding of all parts of the robot. Also everyone should implement a major part of the system and contribute towards integration. I think the grading criteria is pretty spot on.

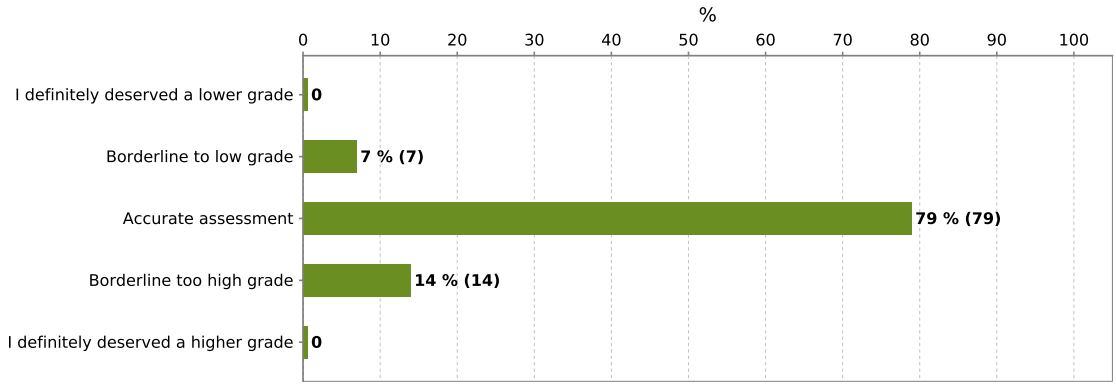
However, for us it was always a bit unclear how exactly the team grade would come about. The only thing we were sure about was that we would get half a grade deducted for missing a milestone. All other criteria were a bit vague.

But was hard to have a feeling whether I was reaching the criteria

I am impressed on how you managed to evaluate the actual work of the individual student in a group project

---

Do you think that the grade you got at the end of the course was a correct asssessment of your fulfillment of the gradig criteria?



Answer choice	I definitely deserved a lower grade	Borderline to low grade	Accurate assessment	Borderline too high grade	I definitely deserved a higher grade
Distribution	0%	6,9%	79,3%	13,8%	0%
Number	0	2	23	4	0

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

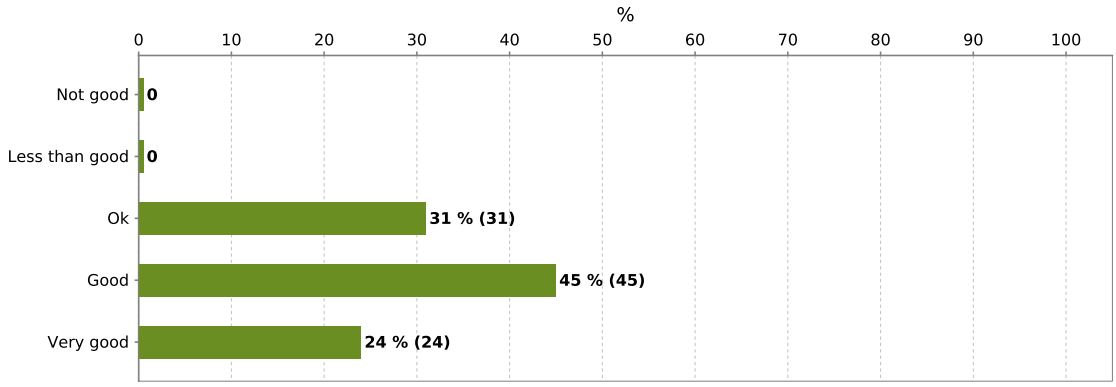
I don't think that the assesment was fair as you graded us based on the reflections but you did not ask the people what they actually think.

I was very happy to receive the feedback that I did. Granted, it was all positive feedback, so it might be that I just like feeling good about myself. However, I believe that the assessment of my role in the project was accurate, precise, and honest.

Yes I feel it was the correct grade, I did implement a major part of the system and definitely did my fair share of programming, but didn't work as much as i should have on integration

---

What do you think about writing the self/group reflections?



Answer choice	Not good	Less than good	Ok	Good	Very good
Distribution	0%	0%	31%	44,8%	24,1%
Number	0	0	9	13	7

29 have answered of 43 (67%)
Maximum number of choices: 1

Respondents comments:

Good way to rethink the group work and the own contribution.

It was ok but I find that your grading was mislead it

I think people underestimate the group dynamics coming into this course. It is important for a group to formulate a shared goal, and for its participants to feel that they play a role in the group's ability to achieve those goals. In this, it is instrumental that people with less skills are given meaningful and adequately-sized tasks. The reflections are a good way to report this, and to achieve shared understanding within the group.

Sometimes it is needed to hear the story from each person's perspective and in discrete. I guess it was important to have that.

Definitely a good chance to evaluate where things went right/wrong, what could be improved in the future, sort out grievances etc.

I think it's very useful for everyone

Things you liked about the course?

Text answers:

Playing with real hardware and solving difficult integration task was a good experience.

Building a robot from scratch and programming it!

the first part was ok but then I started to hate this course.
You drowned us in work and then we did not get help nor more time.

Fuckin' robots man

Weekly progress reports;
Hardwares: RPLidar and UArm;
Milestone specifications;

- Equipment provided
- Project scenario and test environment
- Patric and Diogo
- Fun lectures
- Grading system
- How well the course is prepared and structured

Challenging and got a lot to learn.

We actually got to do a meaningful work with things we learned.

We had great scale of freedom. Periodic evaluation and feedback.

I liked that it as a chance to put what I had learnt into practice. It was very interesting and fun but also very demanding. I definitely don't regret doing the course though!

You learn a lot about cool things and get new friends.

Everything

It was fun! It involved both individual and group work. Very good support by the TA's and Patric. Answers were given very quickly and also personal meetings were always easy to set up.

Working a realistic project from scratch.

Working with interesting problems: vision, motion planning, localization, obstacle avoidance, path planning, etc.

Good hardware to work with.

Enthusiastic teaching team. Most courses the teachers don't really want to teach and only does it since they have to.

You get a lot of cool components to build your own robot. You get your own spot in a room for several months. The tone among the students. Most things, basically.

Having only 40 students (and a way too small room) meant that you got to know many of the students and created a good atmosphere in the course.
I liked being able to build something completely from scratch and go all the way to the end (and not just some small part of a bigger thing).

- learning everything from scratch
- working with people with some very different backgrounds

17 have answered of 43 (39%)

Things that could be improved about the course

Text answers:

-Don't change the schedule all the time it's very confusing if you have other courses.
-Make it even clearer that you should not really have any other courses if you want a live.
- No semi random groups, have the students at least some influence over it.
- Don't make fixed deadline for the milestones. Milestones in itself are fine and necessary but if everyone needs to present at the same time the lab is overclouded and a good examination almost impossible. Also the fixed deadline incentive quick and dirty solution that just barely make the cut for the milestone and need to be redone later in the project.
If the students could present the milestone when they have actually completed it this problem would go away, also a better examination would be possible.
- The penalty for the milestones are very dangerous for the team. You can have the same grading scheme but fulfilled milestones give you points toward an A instead of taking them away forever. When a group fails a milestone the incentive to chat up are basically gone. A system with only suggested deadlines and where you work towards a good grade and a finished system makes more sens.
- A system to deal with problems in the group. Also ask people very directly what they did and let them explain it to make sure they did the work and not another team member.
- Before making a final grade make sure to talk to the person when unsure about some aspects. This can prevent a lot of unnecessary stress and resentment as a wrong assessment can be very damaging to a student.

The work should be divided more equally between periods.

-Do not change the fucking requirements all the time and make sure that the TAs have the same point of view when letting teams pass the milestones

- the TAs had different ideas of what is good enough and some groups got more information than others regarding booby trap, etc....

- the TAs should also stick to what they said and not change from 'oh yes that is good' to 'oh it is not good enough'

-The exam is sooo unnecessary and just stole time that could have been used for way more important exams, especially the grading process...

-Make sure hat the group members do not have to many subjects in parallel, or make bigger teams. Then actually 2 people could work on vision or particle filter and not a whole group gets stuck if one person has a full schedule

- let the groups decided when to present a milestone, just give them some rough guidelines, when approx. they should have done something.
Then the maze is not so crowded when presenting, the TAs will have a better overview and it avoids last-minute-hack-solutions just to pass a milestone

- I think it would be good for the motivation if you would change the grading such that you have some requirements to pass the course and afterwards
you have to or can climb up the ladder but in the pace of the group. So there is no blaming iton one person if his code does not work in time...also everybody has a different schedule and punishing a whole group (by reducing the grade) just because some people have a full schedule is not fair, especially as YOU made the groups.

- check what each single person did for a milestone, so it is not so easy that people claim credit for work they haven't actually done!

I think the booby trap was a little bit too ambitious. It would be better if there was some wall that had springed hinges on it so that you could navigate through it (a la Zelda) and get to some secret compartment or shortcut. It would be more feasible, planning-wise.

The lab room is too small for 9 teams to stay at the same time.

- Lab is too small, there is not even enough oxygen time to time, claustrophobics should be awared (LoL)
- The way the teams are formed is optimal. But the skills tests should definitely be updated.
- Arm should be replaced.

Better grading criteria.

maybe change the exam form, it's really hard to pass

I think at the beginning of the course all the specifications should be 100% clear and then there should be no modifications or additions throughout the course. For example our year had an arm on the robot and it wasn't stated that the height of the robot didn't include the arm so we were trying to find ridiculous ways of making it as small as possible meaning a lot of our components were on the bottom. This was then a huge pain throughout the rest of the course. Also It wasn't known what the objects would be till quite a lot later on. I had an idea of what they would be from the previous year. But if I had known exactly what they were then I might have taken a different approach. Some sort of QR code booby trap was added towards the end of the course, this should definitely have been specified right at the beginning.

Apart from this I would say try and squish the first part of the course so that there is more time for programming the robot.

Other than that I think everything else about the course was pretty much spot on and couldn't have been improved.

It should be like 20 hp instead of 9 hp. I would have liked it a lot better that way since a lot of other courses did have to suffer now.

More credits!

More time, more credits?

Not enough credits for the time required to complete the project. Because of this, I would also add "takes to much time".

The course is too big for its credits, really. The room is too small. I know that you know about these things and that it is hard to improve, though.

I still loved the course, but unfortunately the note on the lab door was spot on - "RAS - Robotics and non-functional systems". The hardware/software we used, even more when used simultaneously, did not perform very well.

If possible start the project a bit earlier since the first milestone does not require any knowledge from the lectures.
Make the specifications for the task (e.g. whether we have to store objects on the robot or not) fixed earlier on in the course since it might affect decisions taken.
Have at least a day between milestone presentation and submission deadline for the videos so that the videos could be filmed during the presentation and then uploaded.

Dont waste so much time before and after lab 1

- Too many meetings?

18 have answered of 43 (41%)