1 – Course Introduction

Robotics and Computer Vision (BPC-PRP)

Course supervisor: Ing. Adam Ligocki, Ph.D.

Ing. Adam Ligocki, Ph.D.

Brno University of Technology
2026

Robotics and AI

Profile

Robotics and AI

Ing. Adam Ligocki, Ph.D.

Position: Assistant Professor

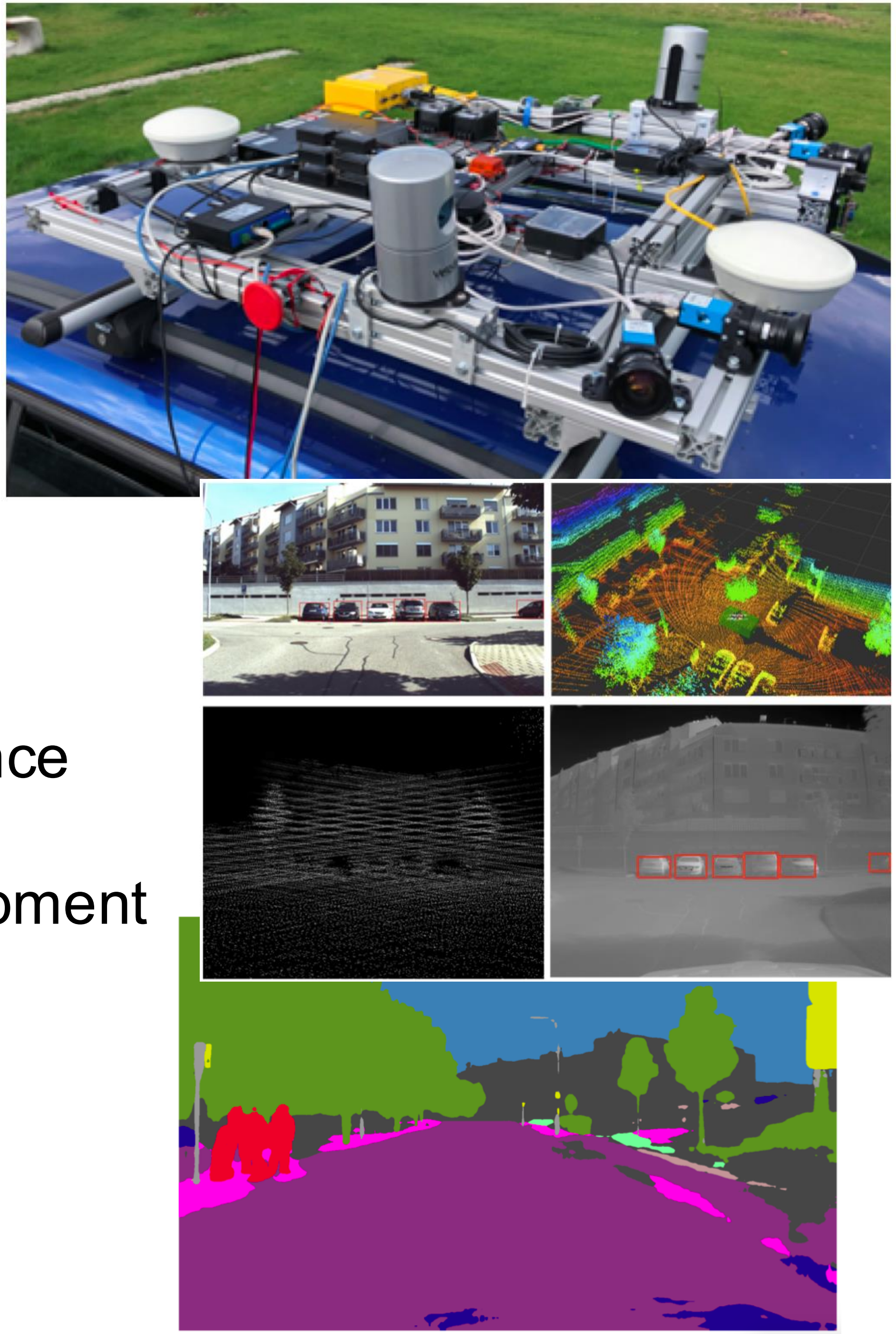
Research: Data Fusion

Room: SE1.102

Web: https://www.vut.cz/lide/adam-ligocki-154791

Background:

- Artificial Intelligence
- Neural Networks
- Software Development



Course Overview

Instructors

Ing. Adam Ligocki, Ph.D.
adam.ligocki@vut.cz, SE1.102

Ing. Petr Šopák
Petr.Sopak@ceitec.vutbr.cz

Course Overview

Welcome to the Robotics course! Over the next 12 weeks, you'll learn:

- **Buildup Basics of Robotics**: Improve your current robotics understanding.
- **Controlling a Simple Robot**: Programming motors and managing movement.
- **Handling Basic Sensors**: Working with line sensors, IMU, lidar, camera, etc.
- **Team Cooperation:** you will work in two-person teams.
- **Evaluate Your Skills in Various Tasks**:
    - Following a line.
    - Driving through a corridor.
    - Escaping a maze.

- **Integrated Challenges**: Applying all the knowledge and skills gained to solve complex tasks.

Course Structure - Semester

The course spans **12 weeks**, divided into key learning sections:

- **Introduction to Basics** (Weeks 1–3):
  - Organization, lab setup, Linux basics, C++, IDE usage, Git, unit testing, and essential tools.
- **Line Following** (Weeks 4–7):
  - Differential chassis, line sensors, PID control, data visualization, and optimization techniques.
- **Corridor Following** (Weeks 8–12):
  - Advanced sensor integration with LiDAR and IMU for autonomous navigation.

Weekly Schedule:

- **Lecture (2 hours)** – Optional but recommended for foundational theory.
- **Lab (3 hours)** – Mandatory, hands-on robotics experiments and implementation.
- **Homework (8 hours)** – Mandatory, focusing on practical assignments and problem-solving.

Course Timeline

Lectures

1. Introduction, Linux, C++
2. Git, Cmake, C++ project, Utests
3. ROS (basics)
4. Motors, Kinematic, Odometry
5. Line sensor
6. Control loop
7. ROS (advanced)
8. Robot‘s sensory equipment
9. Computer Vision - 1
10. Computer Vision - 2
11. Computer Vision - 3
12. Open lecture

Labs

1. Introduction, Linux, env, C++
2. IDE, C++, Cmake Git, Project template
3. Data capture & Visualization
4. Motor control, Odometry, Gamepad
5. Line estimation
6. Line following (PID)
7. **Test: line following**
8. LiDAR - moving in corridor
9. IMU utilization
10. Camera & computer vision basics
11. Individual work
12. **Test: corridor following**

Exam: Date to be estimated

Course Structure – Half Semester Test & Final Exam

During the entire course you can earn up to 100 points.

Half Semester Tests

- 7th week - line following (20 points)
- 12th week - corridor following (20 points)

Maze Escape Challenge:

- Date to be estimated (expected 2nd, 3rd and 4th week of the exam period)
- Up to 60 points
  - 50 points for maze escape
  - 10 points for documentation & paper
- Escaping Maze is MANDATORY! (no passing exam without it)

Course Materials

Fundamental Course Materials

- Course Card in BUT IS: https://www.vut.cz/studenti/predmety/detail/279894
- GitHub Pages: https://robotics-but.github.io/BPC-PRP/

Other Resources:

- Thrun, S.; Burgard, W. & Fox, D. (2005), *Probabilistic Robotics* , MIT Press , Cambridge, MA.
- https://github.com/daniel-s-ingram/ai_for_robotics
- https://www.ipb.uni-bonn.de/teaching/index.html
- https://robotics.umich.edu/academics/courses/

Course Philosophy

European Credit Transfer and Accumulation System (ECTS)

ECTS – standart used for comparing academic „volume of learning“.

1 credit ~ 25-30 hours of student work (for average student)

60 credits ~ 1500-1800 hours (per year)

BPC-PRP -> **7 credits -> ~175 hour** of work per student

Team of two members -> **~350 hours per team**

Average student:

- Lectures: 13 x 2 = 26h
- Labs: 13 * 3 = 39h
- Exam: ~5h
- Home studies ~105h -> ~8h per week

Team of 2 students -> excpected ~16 hours per week

Courese Phylosophy: Open Minds, Support & Responsibility

An Open-Minded Learning Environment

- This course encourages creativity, exploration, and innovation.
- Making mistakes is part of the learning process—don't be afraid to experiment!

Teacher Support

- Your instructors are here to guide, support, and help you succeed.
- Feel free to ask questions, seek clarification, and share your ideas.

Commitment to Results

- While we value learning and growth, achieving results is essential.
- Effort, dedication, and consistent work are key to mastering robotics.

What We Expect from You:

- Engage actively in lectures, labs, and assignments.
- Embrace challenges and treat mistakes as opportunities to improve.
- Take ownership of your progress and strive for the best outcomes.

External Sources Usage

Embrace Learning from the Internet

- This is a **robotics programming class**—searching for and using code examples from the internet is a valuable skill.

Key Rules for Using External Code

Understand What You Use

- If you include code from external sources, you must **fully understand how it works** and be able to explain it.

Respect Licenses

- Follow the licensing terms of the code you use.
- Cite the original author and provide proper attribution where required.

No Peer Copying

- Copying code from your colleagues (current or past BPC-PRP courses) is **strictly forbidden**.

Why This Matters

- Programming is about learning and growing—not just copying.
- Respecting licenses and ideas fosters professionalism and integrity.

Can I use ChatGPT?

ChatGPT, Claude, Cursor, Copilot, …, are a Tool, Not a Shortcut

- Using ChatGPT to brainstorm ideas, generate code, or seek explanations is **allowed**.
- However, its use must align with academic principles and ethical standards.

Key Rules for Using ChatGPT

Understand and Verify Outputs

- Ensure you **understand** and can explain any material or code generated by ChatGPT.
- Verify its accuracy and adapt it to your specific context.

Acknowledge its Contribution

- Clearly cite ChatGPT if you use its output in your work, just like any other academic source.
- Example: *"Code/idea generated with the assistance of OpenAI's ChatGPT."*

No Substitutes for Personal Effort

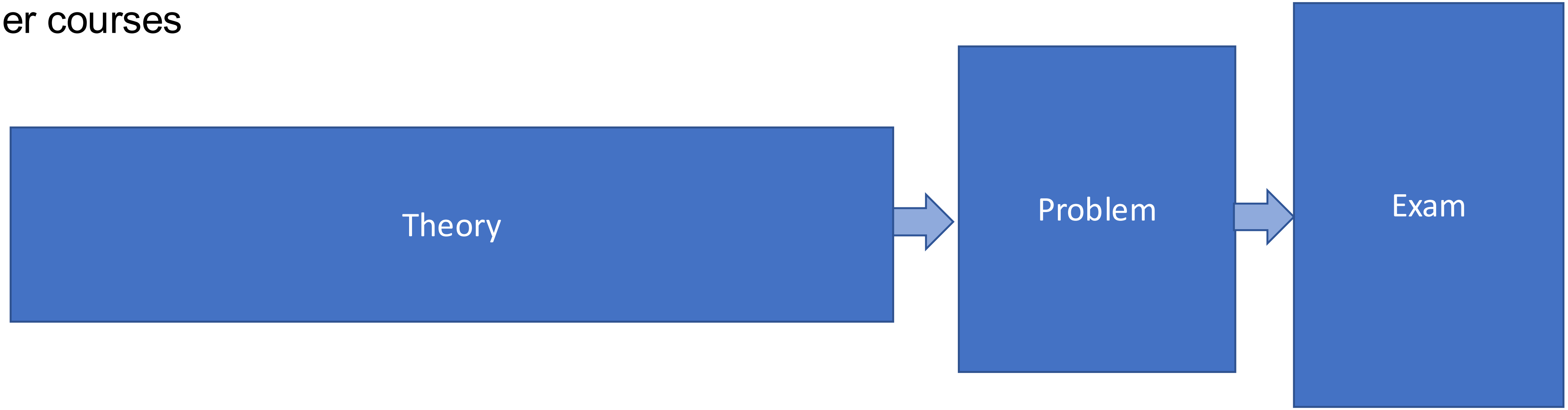
- ChatGPT can help, but your **own effort, critical thinking, and creativity** are essential..

What's Not Allowed

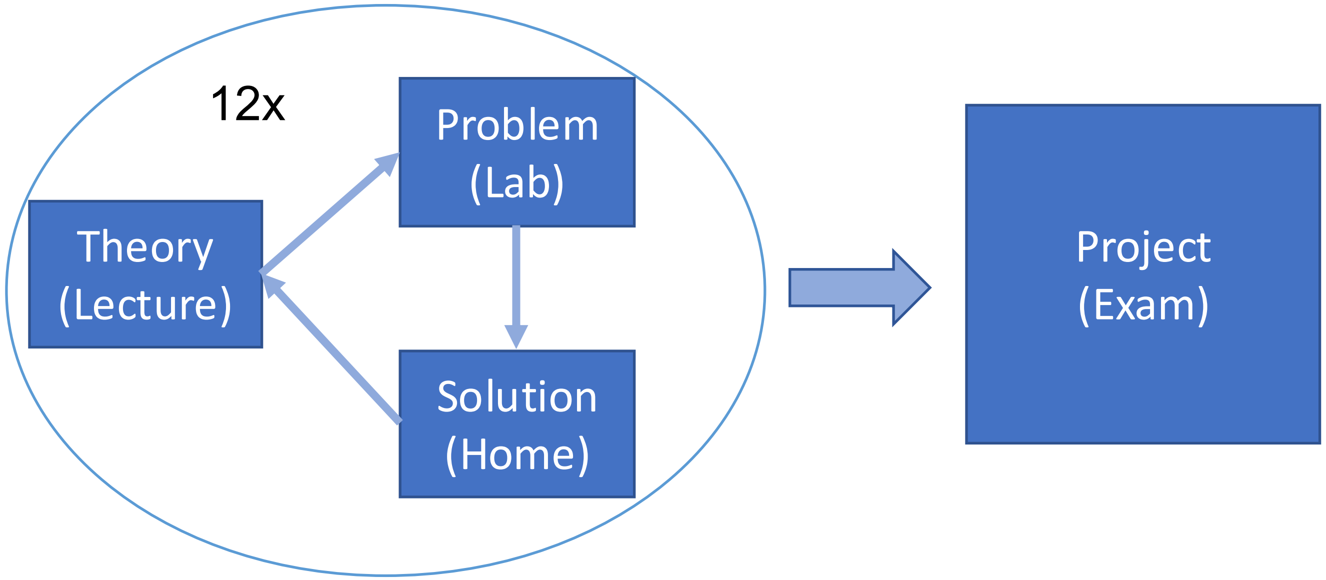
- Using ChatGPT to bypass the learning process or submit its output as if it were entirely your own.
- Any misuse that violates the university's academic integrity policies.

Lorem Ipsum

Other courses



BPC-PRP



Your Feedback

We've put significant effort into designing this course to deliver the **best student experience**. However, we know that **not everything is perfect**—there's always room for improvement.

Help us Improve

- If you notice any **mistakes**, find something **hard to understand**, or think an **explanation is unclear**, please let us know.
- Feedback isn't just for the end of the course—feel free to share your thoughts **anytime** during the semester.

How to Provide Feedback

- Use the official course feedback form or talk to us directly.
- For sensitive or anonymous input, you can use an **anonymous email**.

Your Feedback is Valuable

- **Negative feedback is as important as positive feedback**—it helps us make the course better for everyone.
- We appreciate your honesty and willingness to help us improve.

Student‘s Feedback from 2025

1.Practical focus and motivation

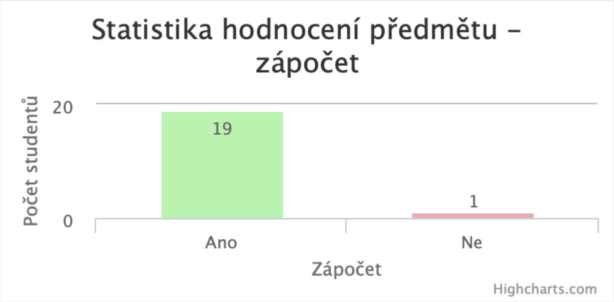
“I was rarely this passionate about studying; this course captured exactly why I enjoy studying in the firs place.”

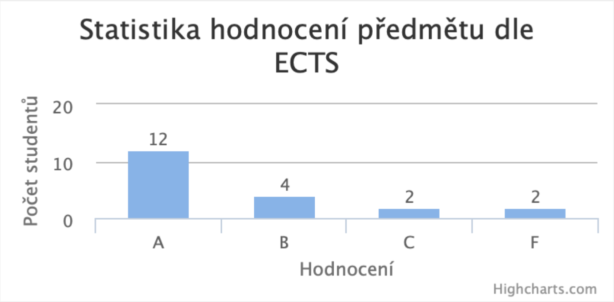
2.Real knowledge gain

“In three years of studying, this was the first practical course where the result of my work was a truly functional system.”

3.Workload vs. benefit

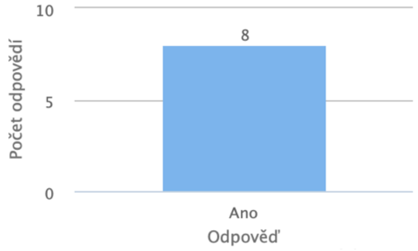
“The course is time-consuming, but interesting. In project-based courses I learn far more than from traditional exam-focused memorization.”

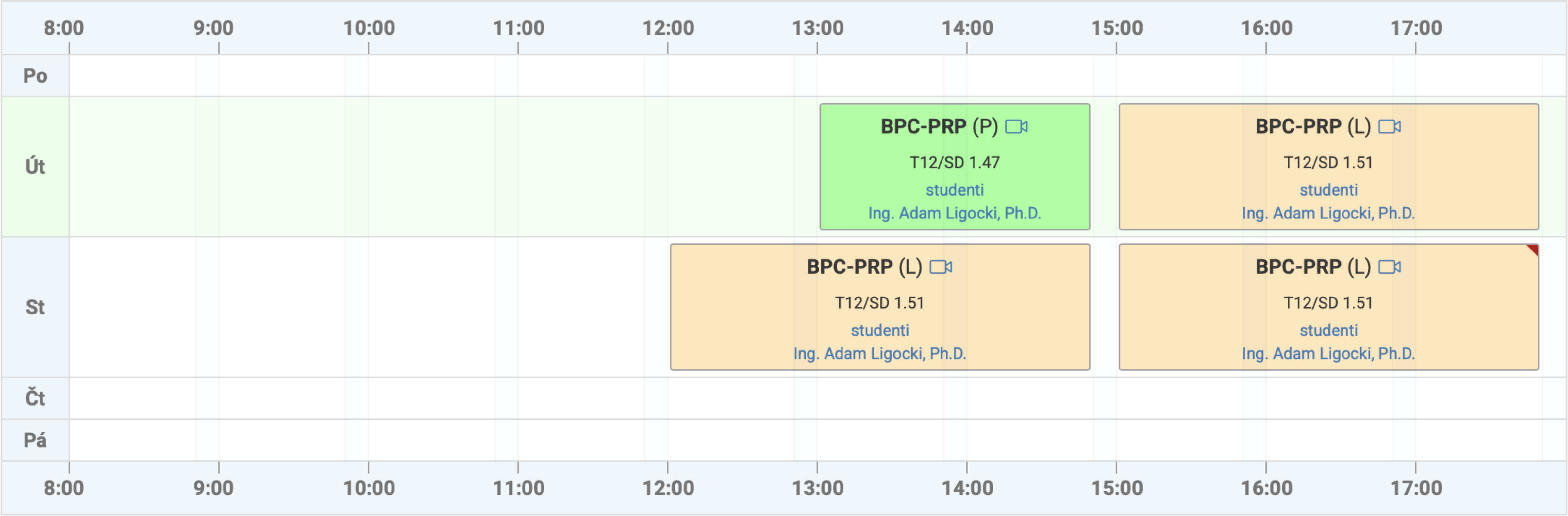




1. ZAUJALA VÁS PŘEDNÁŠENÁ PROBLEMATIKA A POMOHLA VÁM PROBÍRANÁ LÁTKA K ORIENTACI V DANÉ OBLASTI?

Odpověď	Poč. odpov.
Ano	8 (100 %)

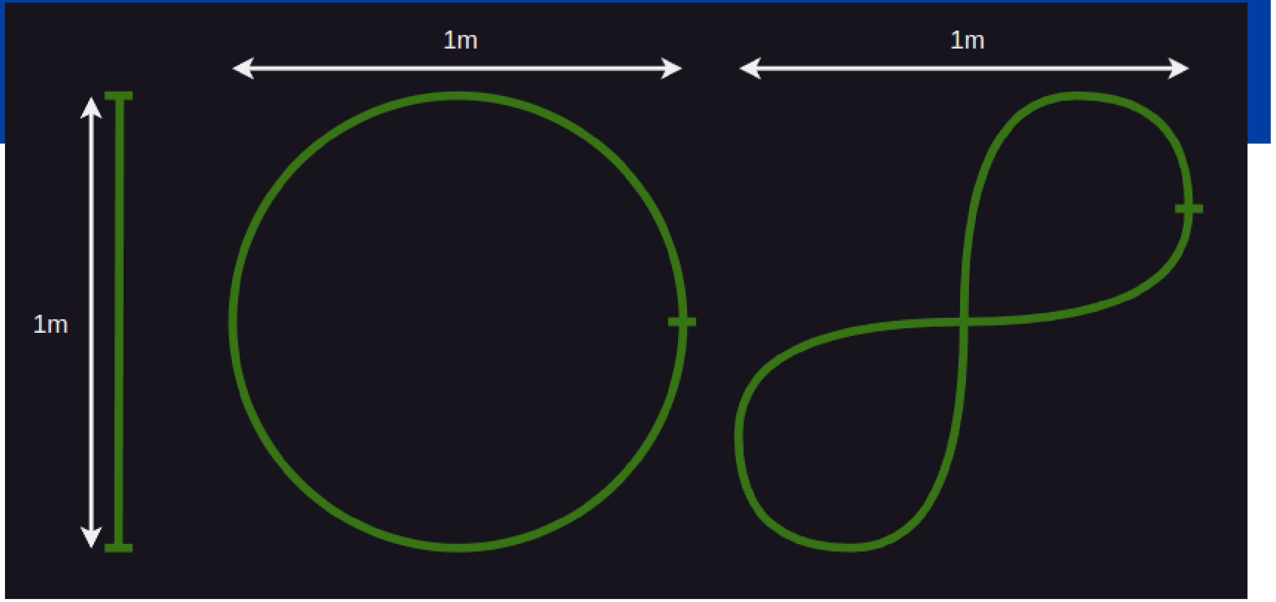



8:00
9:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
Po
Út
BPC-PRP (P)
T12/SD 1.47
studenti
Ing. Adam Ligocki, Ph.D.
BPC-PRP (L)
T12/SD 1.51
studenti
Ing. Adam Ligocki, Ph.D.
St
BPC-PRP (L)
T12/SD 1.51
studenti
Ing. Adam Ligocki, Ph.D.
BPC-PRP (L)
T12/SD 1.51
studenti
Ing. Adam Ligocki, Ph.D.
Čt
Pá


ORIGINAL
PRUSA

Half-Semeter Tests & Final Project / Exam

Line Following – 8th Week (25p)



There are 3 tracks:

- Straight Line (5 points)
- Simple Loop (5 points)
- Complex Loop (10 points)

To pass the track robot has to follow the full length of the line.

If no part of the robot's body covers the line, attempt fails.

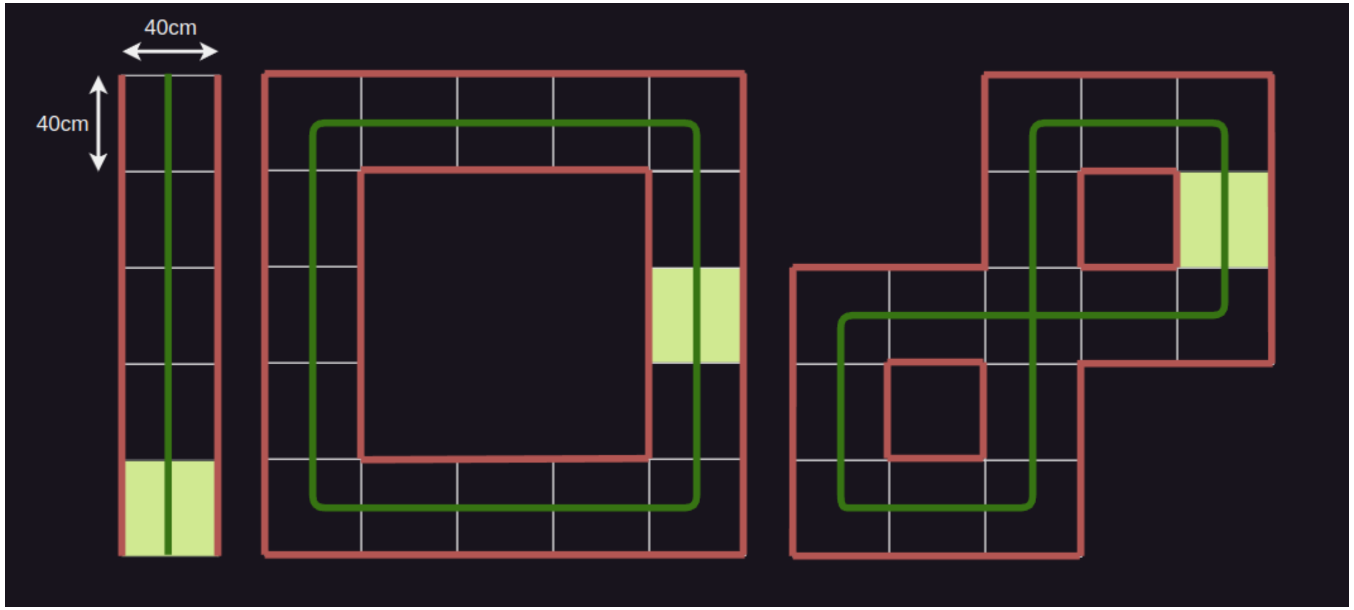
The points are awarded only for completing the entire track.

Teams have 3 attempts per track, with a time limit of 3 minutes per attempt.

All 3 attempts have to be performed during a single lab.

Rules details: https://robotics-but.github.io/BPC-PRP/2_labs/text/8_lab.html

Corridor Following – 12th Week (25p)



There are 3 tracks:

- Straight Corridor (5 points)
- Simple Loop (5 points)
- Complex Loop (10 points)

Corridor is defined by walls.

All tracks are in rectangular grid of 40x40cm

Cells are marked by black tape on the ground.

The points are awarded only for completing the entire track.

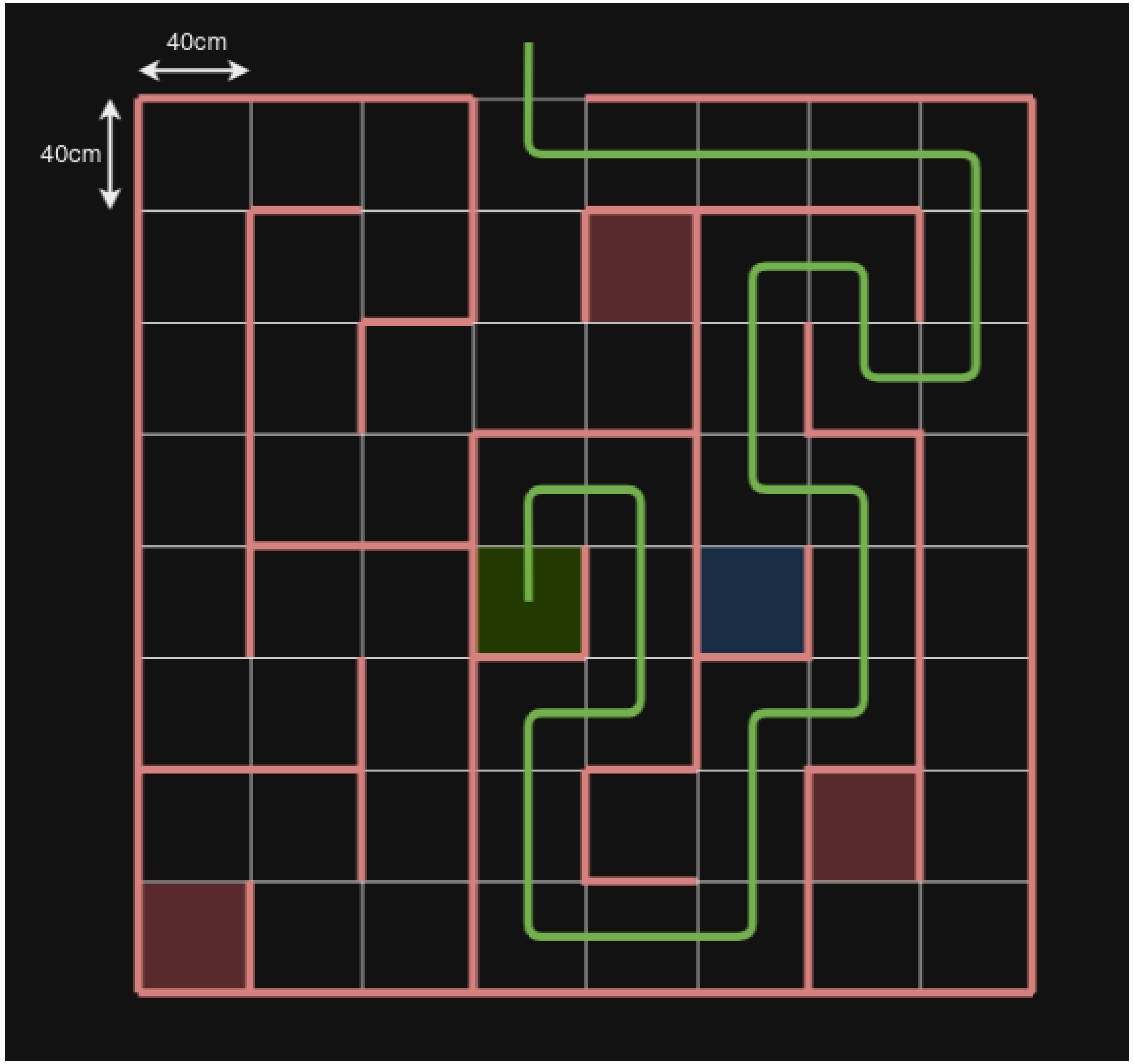
- pass straight corridor without touching the wall
- pass full loop
- no wall touch allowed
- no multiple cell entrance allowed

Teams have 3 attempts per track, with a time limit of 3 minutes per attempt.

All 3 attempts have to be performed during a single lab.

Maze Escape (Final Exam) – Exam Period (50p)

- 8x8 Maze, cell 40x40cm
- Up to 50 points based on escape time
  - p = min(max(kx + q), 0, 50)
- 3 Minotaurs (+30s) and 1 treasure (-30s)
- Wall touch (+10s), wall destruction (DNQ)
- No loops, one escape path
- Aruco tags shows escape paths and treasure escape path
- Each team has 3 attemts
  - Limited time for 1 attempt (5min)
  - 45 mins between attempts
- Code used for race will be in online Git repo



Rules details: https://robotics-but.github.io/BPC-PRP/2_labs/text/13_exam.html

Key Skills

Real C++ Cmake project with Gtest + GIT

ROS2 Environment

Motor control

Line following using Line Sensors

Corridor following using LiDAR

Utilizing IMU for orientation estimation

Camera & Aruco tags

Integrate all in one to escape the maze

Project Documentation (10p)

Git Repository Quality

- Clear commiting policy
- Well documented commits
- README.md
- Well structured code

Short Paper

- Write a 2 A4 paper, that briefly describes your project
  - Overall working principles
  - Highlight challenges and successes
  - Something interesing
- Put paper into Git repository to **/paper** folder
- Use template from https://github.com/Robotics-BUT/BPC-PRP/tree/master
- Recommanded to write paper in https://www.overleaf.com/ tool.

Overall Course Evaluation

Half Semester Tests:
Up to **40 points** from two practical demonstrations of task solutions during the exercises:

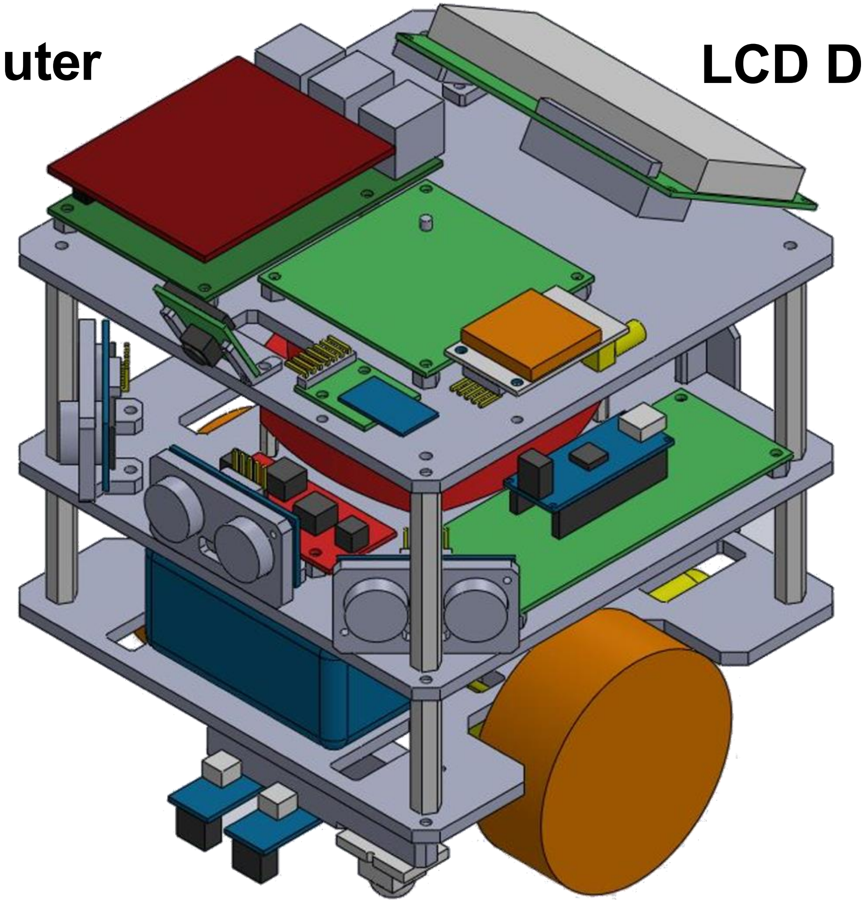
- **20 points** – Robot line-following:
  - 5 points – Straight line
  - 5 points – Simple loop
  - 10 points – Complex loop
- **20 points** – Robot navigation in a corridor (using LiDAR):
  - 5 points – Straight corridor
  - 5 points – Simple lopp
  - 10 points – Complex loop

Final Exam:
Up to **60 points** from the final exam in the form of a “maze escape competition”:

- **50 points** – Maze escape
  - Overall time
  - + Treasure
  - - Wall collision
  - - Minotaurs
- **10 points** – Project management (Git), documentation quality and technical paper.

HW & SW Overview

Robot Overview – Top Deck



Raspberry Pi Computer

LCD Display

IMU

Barometer

ADC

RTC

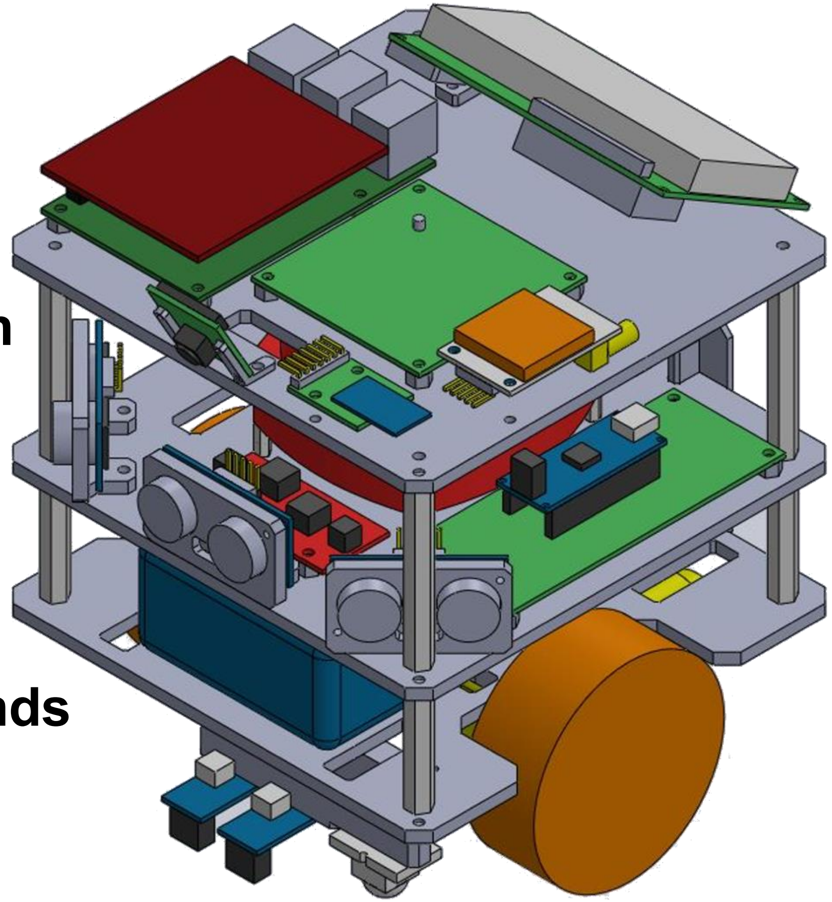
GPS

UWB

Camera

Details: https://github.com/Robotics-BUT/fenrir-project

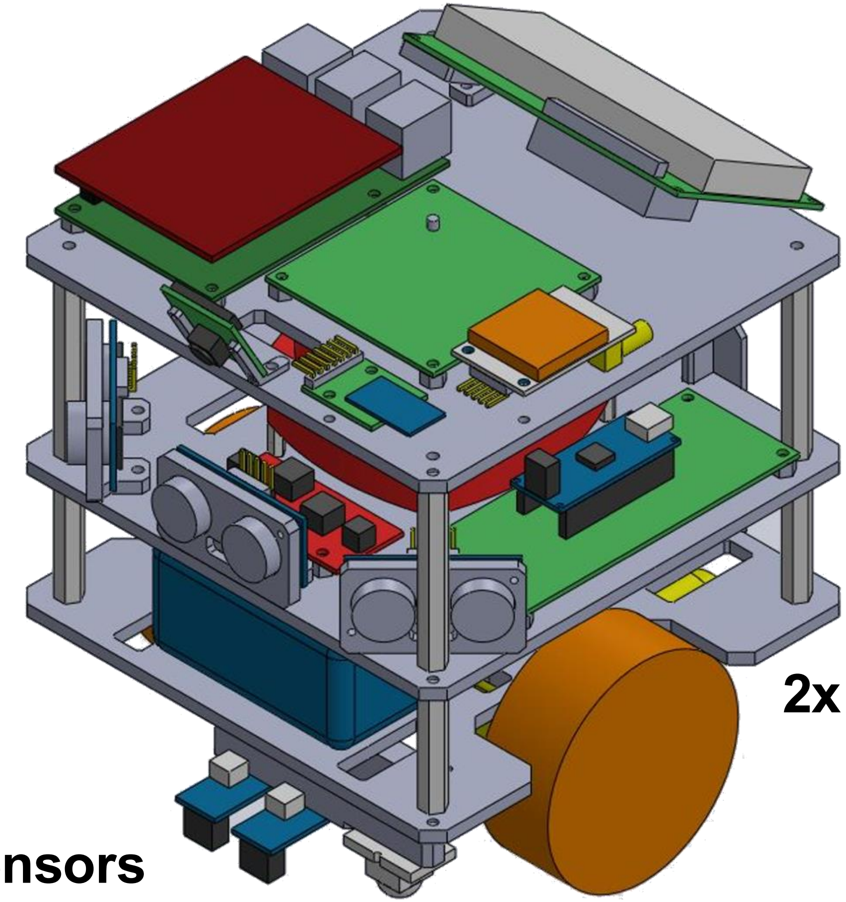
Robot Overview – Mid Deck



Details: https://github.com/Robotics-BUT/fenrir-project

Robot Overview – Downsite

2x Battery 10000 mAh



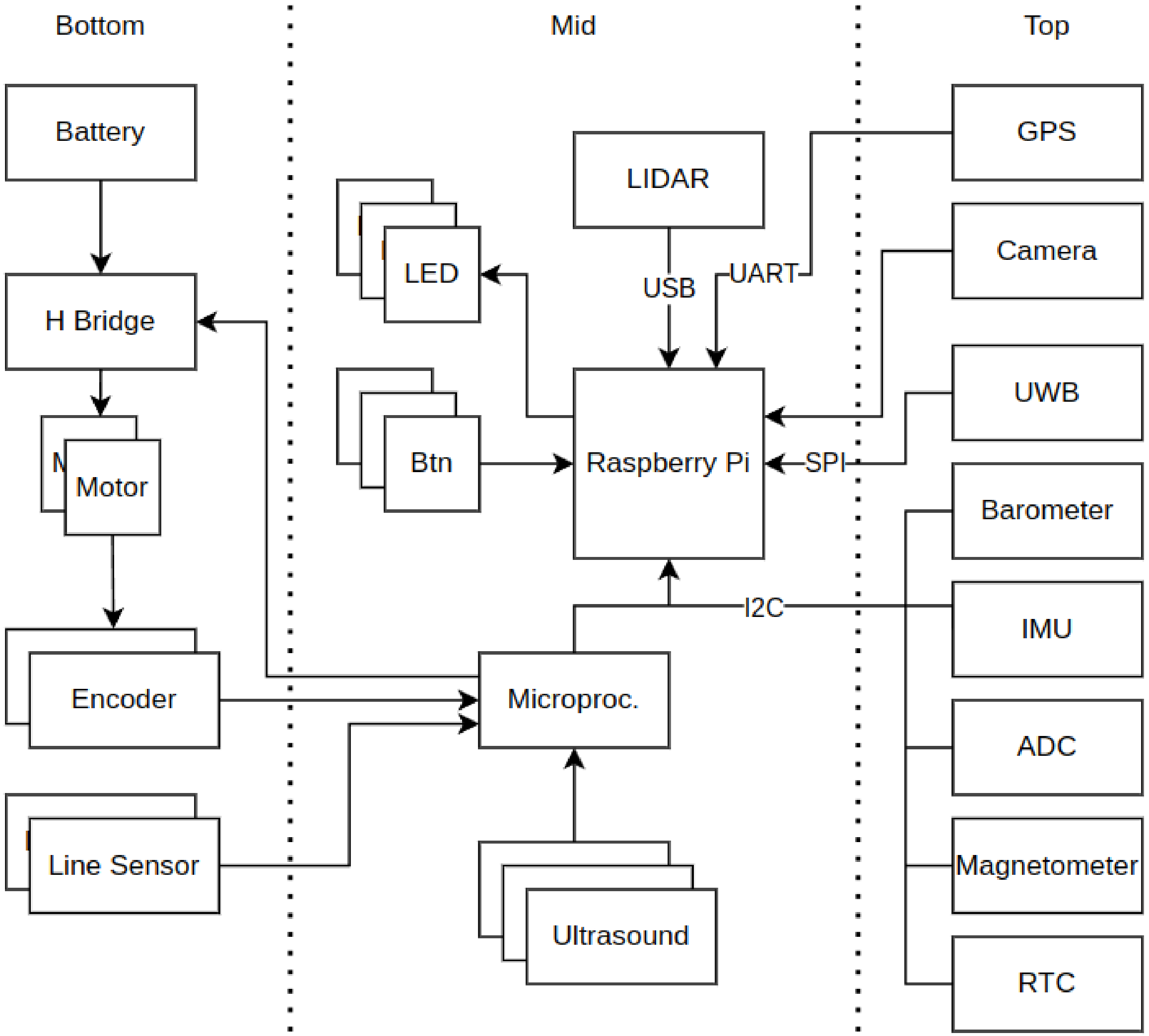
2x DC Motors + Encoders

2x Line Sensors

Motor Driver

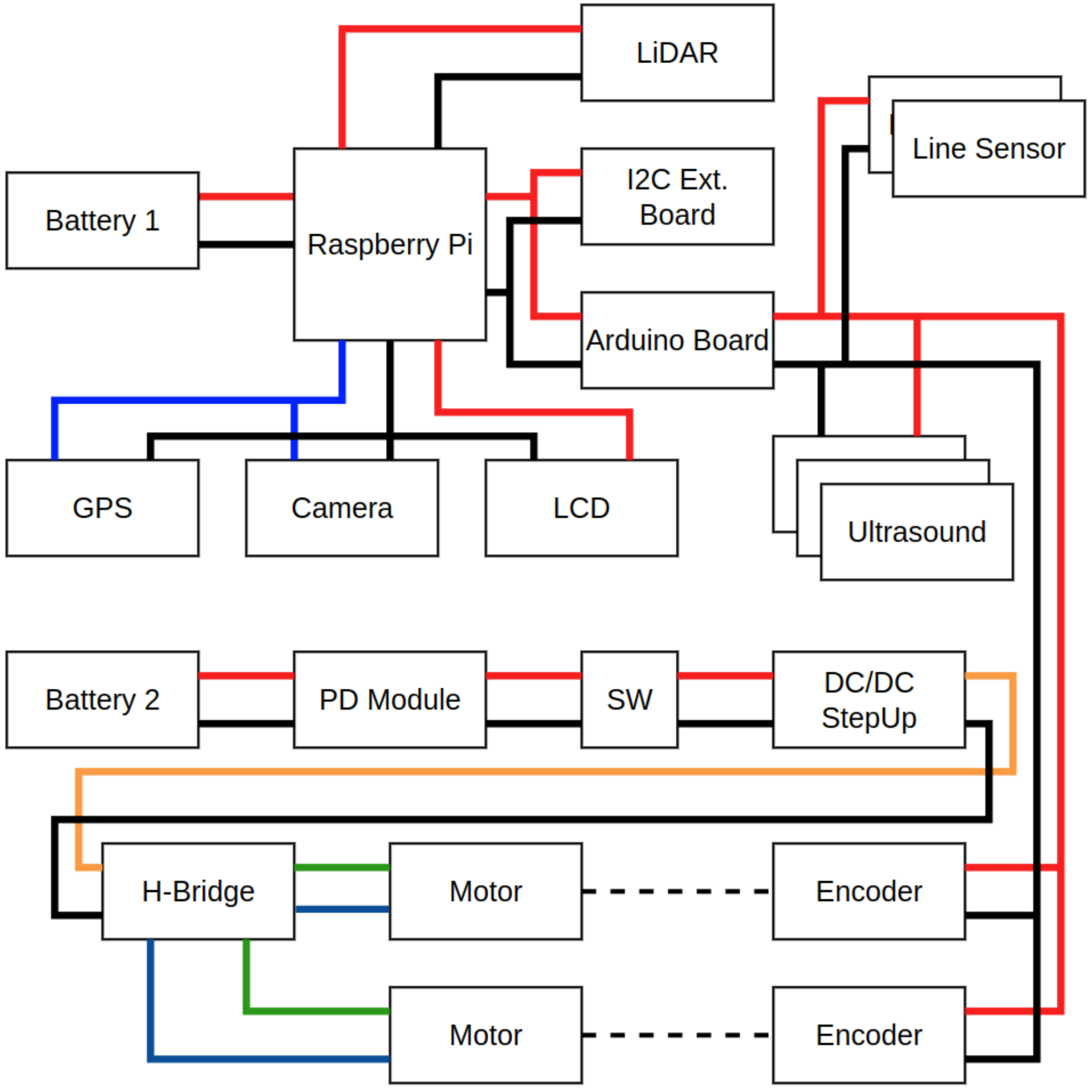
Details: https://github.com/Robotics-BUT/fenrir-project

Robot – Component Scheme



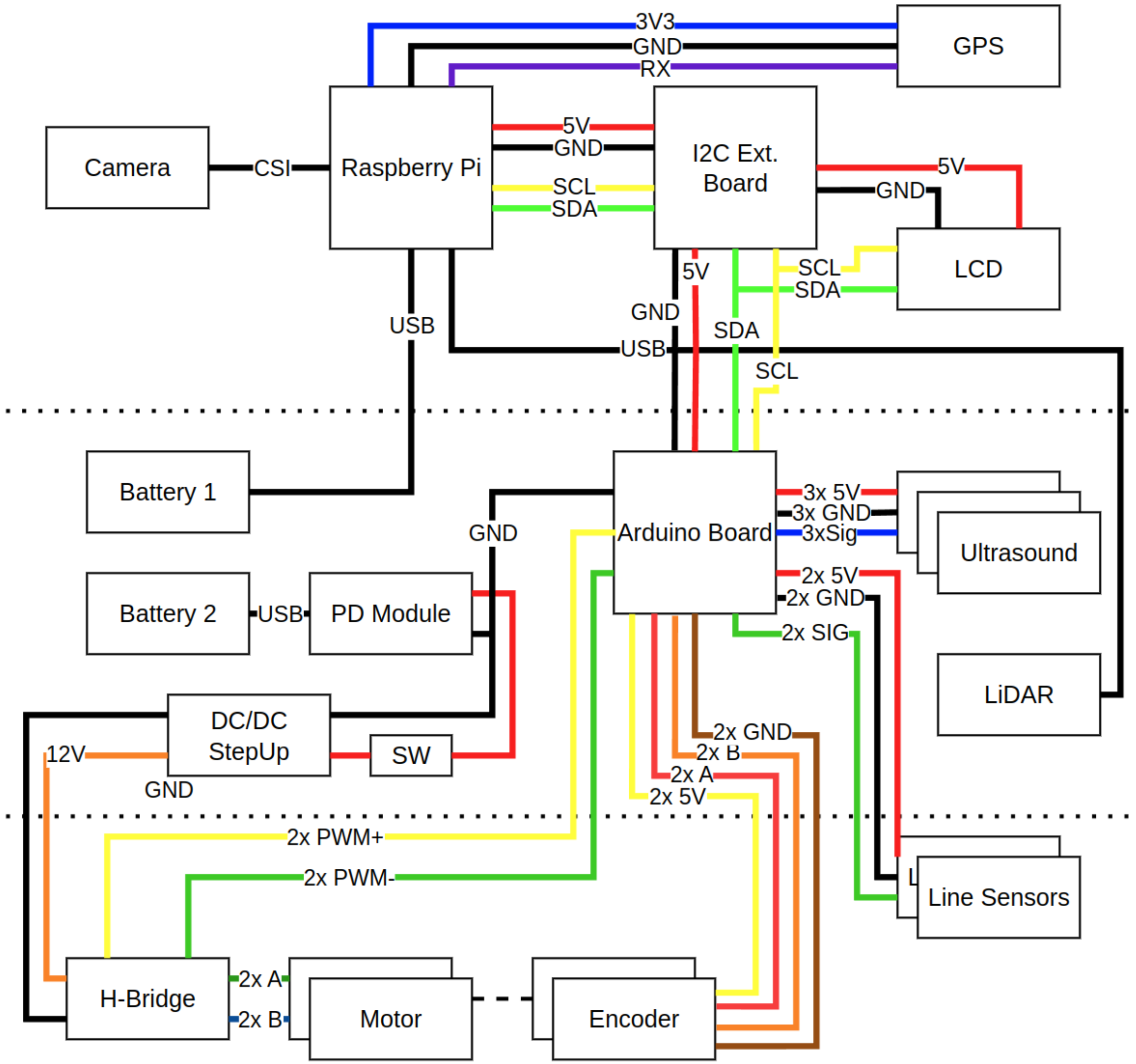
Details: https://github.com/Robotics-BUT/fenrir-project

Robot – Electrical Schema



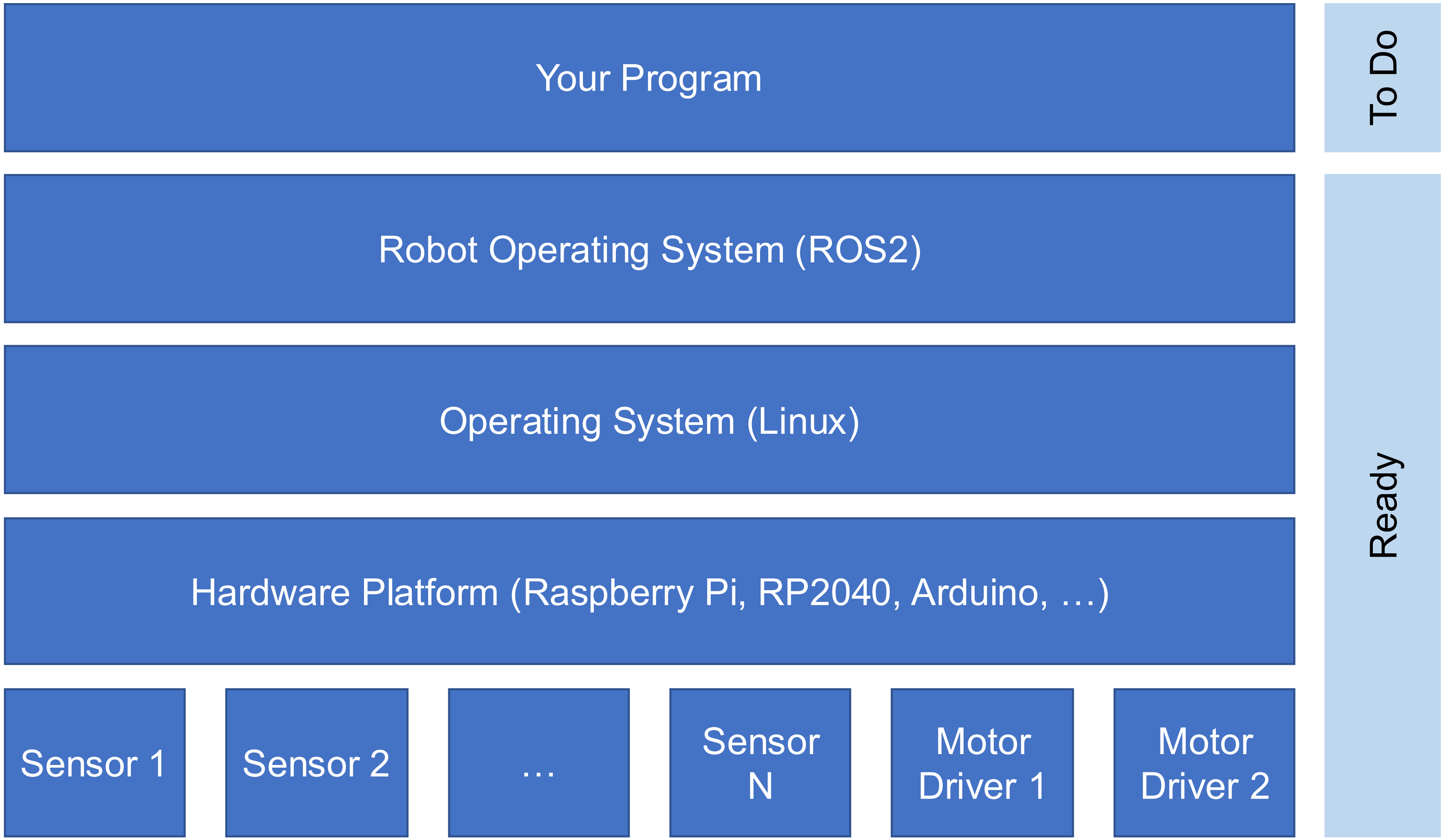
Details: https://github.com/Robotics-BUT/fenrir-project

Robot – Wiring



Details: https://github.com/Robotics-BUT/fenrir-project

ROS2 Abstraction



Adam Ligocki

ligocki@vutbr.cz

https://www.linkedin.com/in/adamligocki/

https://github.com/adamek727

Robotics and AI Research Group

https://github.com/Robotics-BUT

Brno University of Technology
Faculty of Electrical Engineering and Communication
Department of Control and Instrumentation