CU04448 Intelligent Control Course Instruction

Hanshu Yu

1st September 2025

1. Lectures & Labs

Lectures will be divided into 2 large themes:

Blok1: Python programming basics

Blok2: Programming/Data related knowledge in real-life scenarios

Labs will be free-formed self-struggling around the lab assignment. Only the most basic instructions will be provided, freedom is given to student teams.

Time consumption:

28 * 7.5 * 40% = 84 hours per student.

Each team 4 students, so 336 hours of work.

Attendance will *not* be checked, however it is strongly recommended that you attend.

Important details will be in lectures, not all detailed explanations will be on learn because that is too much work. Also, you can ask questions and get answers.

Lectures slides will be published on learn, they are exactly what I'll be using during lectures. They are formulated to be a syllabus like format, if you find yourself able to explain the concepts in the slides without peeking at some reference material, then you are up-to the "very good" standard and can expect an 8,0+ final grade.

Q: Programming work put an emphasis on coding skills but why check theory?

A: These theory & concept contents can only be mastered when you write a lot of code and put things into use. You can obtain the code from whatever source and can be usable, but you will not be able to modify those in the future. Checking these theory & concepts demonstrate that you actually did the coding or actually understand the code.

I am extremely busy in Blok 1&2, expect me to reply to your email within 5 working days.

2. Lecture accompanying assignments (TOETS02)

There are 4 topics available.

One mandatory topic:

- ✓ “The Ancient Imitation Game” – Decoding Caesar Cipher

Choose 2 from 3 free-elective topics:

- Cellular mobile network operator simulator
- PID controller simulator
- Fair or not? Synthetic hiring data exploration

There are also possibilities to do exploratory bonus work for your interested topic. For detailed possibilities & requirements, see the description file for each topic. For an explanation of “exploratory bonus” see section 4: *Assessment and grading (TOETS 01 & 02).*

3. Lab assignment (TOETS01): Indoor Robot Localization

The scenario:

You run a large factory, you have the map of the factory. You deploy multiple small robots in the factory, they reset every night to perform system upgrade and maintenance. However, when the robots reboot every morning, they lose their location information and it’s impossible for these robots to function without knowing their location. It is impossible to localize the robots manually, the robot must localize itself in the factory.

The method:

We utilize the concept of **particle filter**, a Bayesian Monte-Carlo stochastic method to discover the location of the robot.

The assignment:

Implement this method on an robot equipped with Raspberry-Pi. The robot can move in space, either with wheels, continuous tracks, crawlers, or even a drone that can fly. You are free to choose how your robot can move based on your problem analysis & design.

Deliverables:

- *Report*
- *Prototype*
- *Demo*

4. Assessment and grading (TOETS 01 & 02)

We make use of “portfolio-based oral exam”. Your portfolio consists of: *report*, *code*, *demo*.
The basic tasks are designed to be completed with less than 250 hours.
If you do not do any exploratory bonus for TOETS02, then your maximum attainable point is 8.0. In case that a teams’ hardware fails completely for TOETS01 (this happened before), then your maximum attainable point is 8.0. (In both cases PartA + PartB = 8.0)
To reward those who try extra, the total attainable point is 11 but max out at 10. Such that if you are interested to explore more, you do get the reward.

4.1 PartA: report and code: 50% (TOETS 01 & 02 same)

Format (0.5) *Page limit, components in section 2, code in appendix, grammar*	*Remarks*	*Grade*
Report readability (0.5) *Decent writing quality, logical and concise, to the point.*	*Remarks*	*Grade*
Report contents (1) *Clearly presents your analysis and design choice, you can demonstrate your analysis and implementation.*	*Remarks*	*Grade*
Code quality (1.5) *Readable, with just the right amount of comments, no obvious logic loop-holes, no obvious security threats to user, meaningful variable names, re-useable.*	*Remarks*	*Grade*
Code functionality (1.5) *The code work with 3 necessary functionalities defined in section 1. You can demonstrate this with test cases.*	*Remarks*	*Grade*
Score *Max: 5 pts*		

4.2 PartB: interview: 30% (TOETS 01 & 02 same)

I’ll conduct an interview 15-20 min per group. Main content: discuss your work and your relevant knowledge. The evaluation will be based on your individual performance and knowledge.

Justify your work (1.5) *You can explain and justify your work, including report contents and code.*	*Remarks*	*Grade*
Demonstrate your knowledge (1.5) *You should know the basic necessaries about python programming, object-oriented methods, and fine coding conducts.*	*Remarks*	*Grade*
Score *Max: 3 pts*		

4.3 PartC: 30%

(TOETS 01 Robot Demonstration)

Your robot can successfully localize itself with limited move/time(2) *Sufficient (0.6): within 60 moves, or within 7 min* *Good (1): within 25 moves or within 3 min* *Very good (2): within 20 moves or within 1.5 min*	*Remarks*	*Grade*
Q&A handling(1) *Your robot operates the same as what is presumed in your design. If not you can elaborate why. You can demonstrate that you have considered the practical issues thoroughly.*	*Remarks*	*Grade*
Score *Max: 3 pts*		

(TOETS 02 Exploratory bonus)

Dutch language Caesar Cipher (0.75) *You can explain and justify your work, including report contents and code.* *Usable: 0.25* *Explanation & justification of method: 0.5*	*Remarks*	*Grade*
Free-elective exploratory bonus (2.25) *You should know the basic necessaries about python programming, object-oriented methods, and fine coding conducts.* *Usable: 0.75* *Explanation & justification of method: 1.5*	*Remarks*	*Grade*
Score *Max: 3 pts*		

4.4 Final score calculation

MIN{ PartA + PartB + PartC , 10}= Final score toets02

4.5 Additional explanation

PartA: Score report and code (50%) :

the score will be a group score, everyone is the same.

PartB: Score interview (30%):

this one will be individual, based on your performance in the interview.

PartC: Exploratory bonus (25%):

the score will be a group score, everyone is the same.

Additional modifier:

Late submission modifier. See “Rules for exceeding the deadline” below.

5. Rules for exceeding the deadline

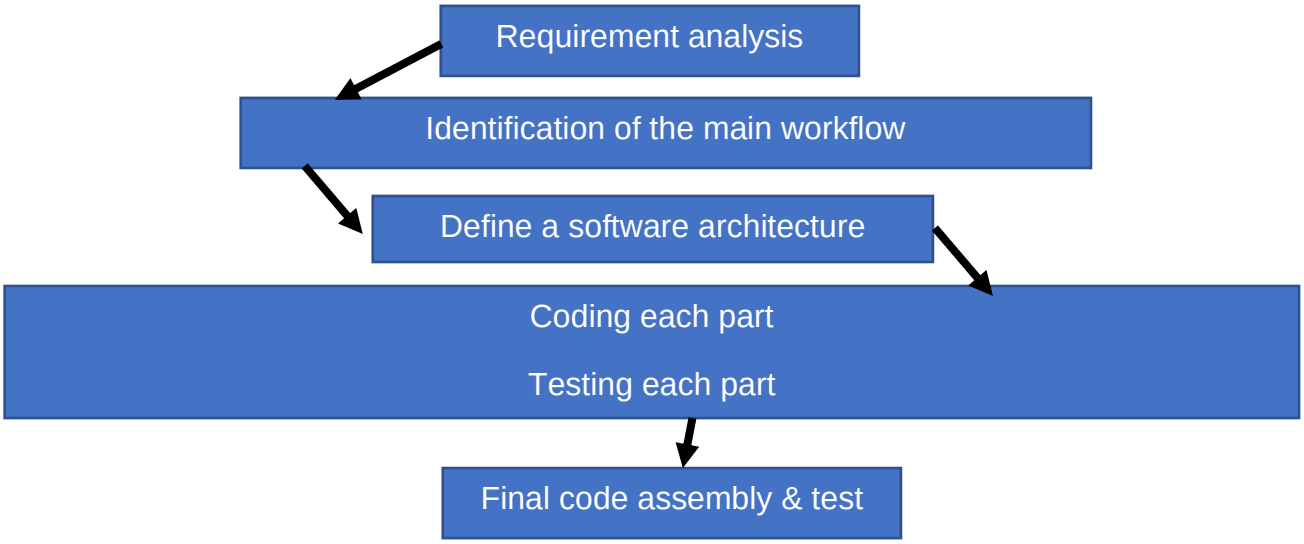
Exceeded hours: *x*

(Always rounded up to the next integer! So, if you exceed at least 1 minute, it's counted as 1 hour! If you exceed 1 hour + 1 minute, it's counted as 2 hours!)

Final score before the deduction: *s0*

Final score after the deduction: score_final = s0 * (1 - (sqrt(x*2)/20))

6. Recommended framework of code collaboration



7. Use of AI tools policy

Allowed:

- Help with study
- Help with code writing
- Help with debugging
- Help with report writing

Tip:

Claude is currently the best in coding tasks.;-P

Consequences:

When you submit your report and code, I will read everything in detail.
I will ask very detailed questions in your report and code.
If you cannot answer the part you include in your report/code, the points corresponding to that part will be set to ZERO in the grading rubric.

8. Typical questions you may expect in the oral exam (BUT NOT ALL)

Chatting:

How are you? How do you like the project? Did you eat?

General questions:

What do you think is the most difficult part of this project?

Which part of Python programming do you like the most/least?

Code:

Tell me which part of code you still feel that you can improve?

Why did you write …… here? What's the functionality?

If I change …….. to ……, what will be the consequences?

How do you handle errors?

How did you debug your code? Are there any funny bugs?

What (method/tool/……) did you use to guarantee the (some quality) of your code?

Report:

Why did you make this choice?

How did you come to this software architecture?

What are the practical matters that you think this problem did not yet address?

What happens if I ……..?

How did you approach & analyze the exploratory bonus part?

What makes is different/difficult?