also known as O1

We'll start at 14:15.

Intro Session, Fall 2023

Intro to the O1 Course and a Programmer's Tools

For Finnish-speakers: Tervetuloa! Tämän salin luento on englanniksi. Jos osaat suomea, siirry mieluummin Kanditalon A-saliin, Otakaari 1. Siellä on suomenkielinen kurssiesittely samaan aikaan.

FAQ answer: If you've already worked on O1's first assignments online on your own, this optional introductory lecture will not teach you much that is new.

presented by Otto Seppälä, otto.seppala@aalto.fi
slides by Juha Sorva, juha.sorva@aalto.fi

Coming Up in This Lecture

1. Course arrangements and practices.

2. Some basics about our actual topic, programming.

This lecture serves as an intro to the assignments that you'll tackle during O1's first week. You can get started with those at any time, even right after the lecture.
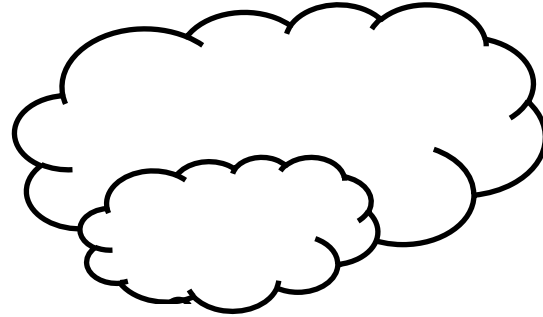
Our Main Goals in Brief

We hope that after finishing O1, you...

... think that programming is a fun thing to do.

... have some practical programming skills.
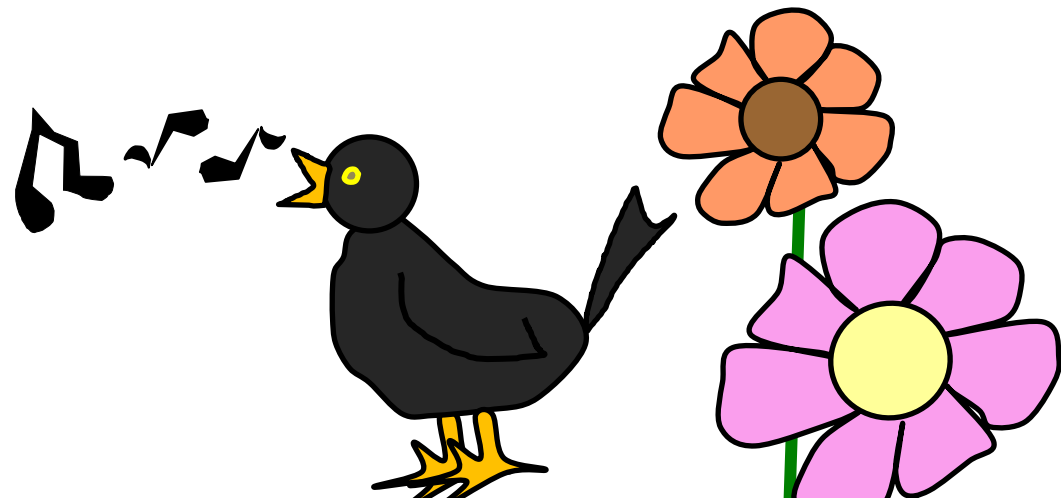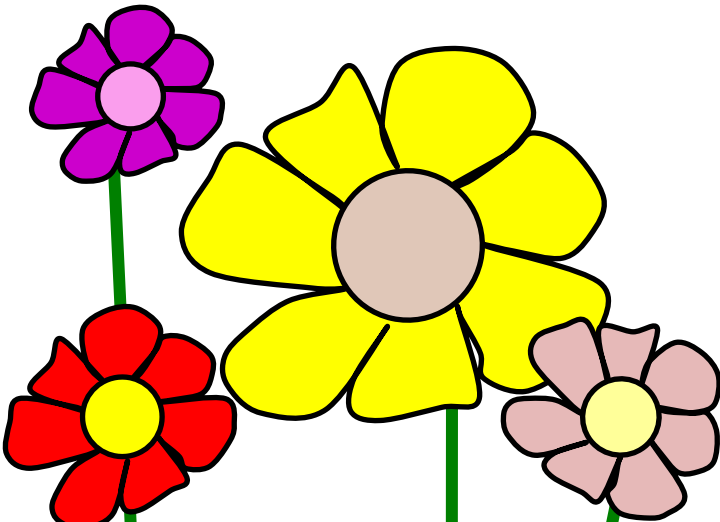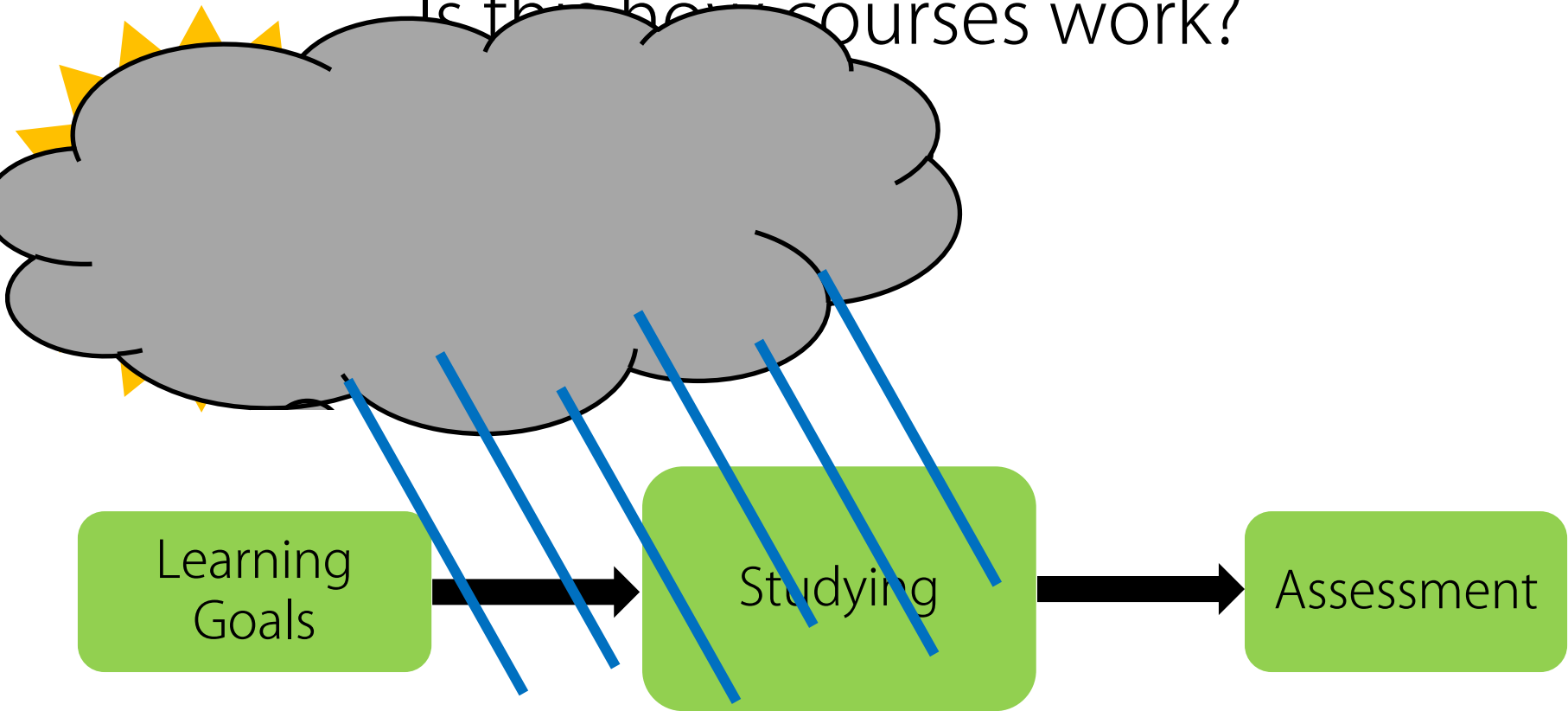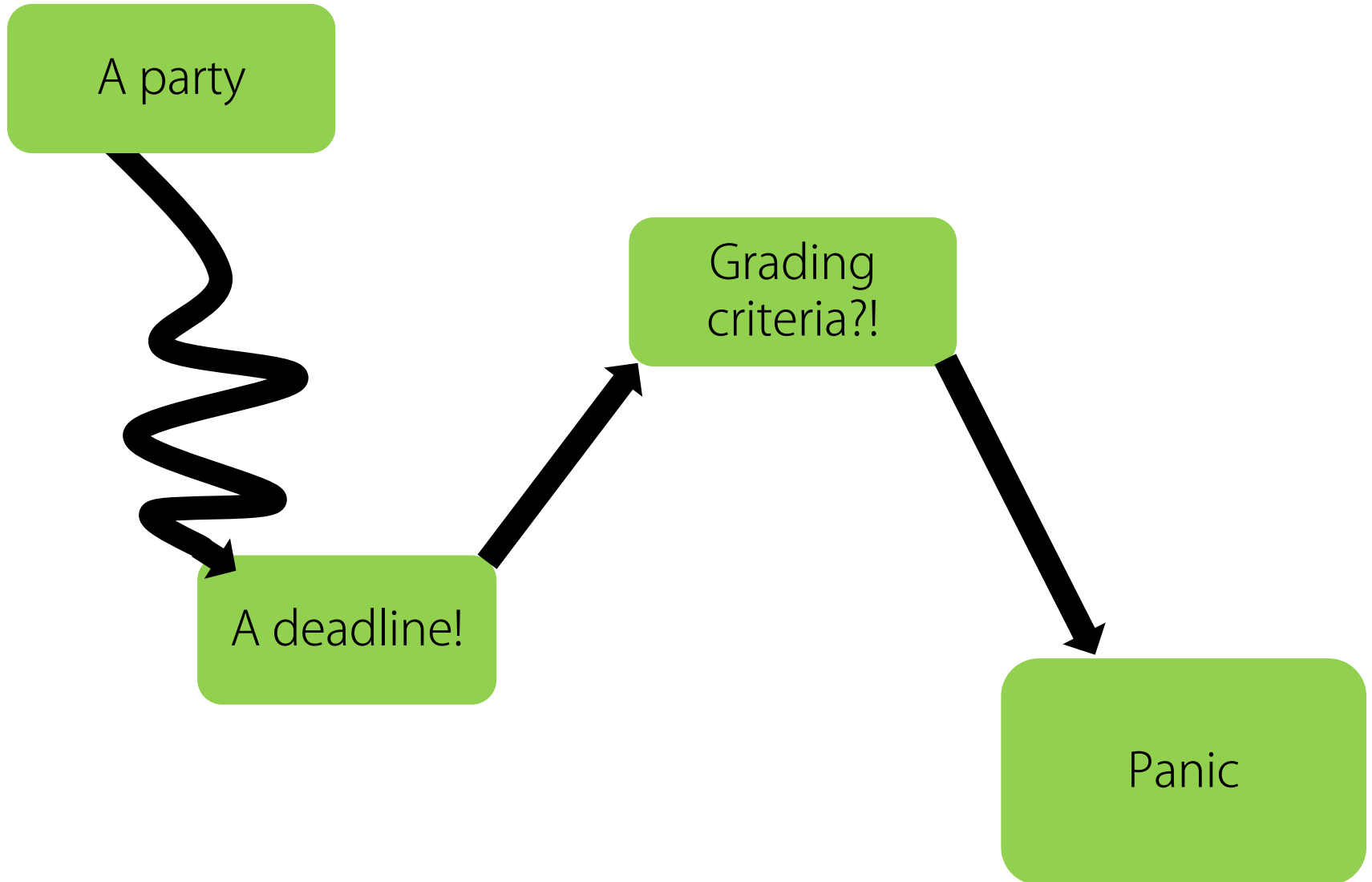
Is this how courses work?

Learning Goals → Studying → Assessment

Is this how courses work?

Learning Goals → Studying → Assessment

Meanwhile in the real world...

About This Course Intro

- We do have nice, specific learning goals in mind, but it's hard to appreciate them just yet, when you haven't yet learned the relevant concepts.

- Let's instead take a look at O1 in concrete terms, starting from assessment and eventually reaching the course contents.

# On Grades and Learning

- We've tried to design O1's grading scheme so that if you pay attention to it, you will:
  - a) learn
  - b) get to do fun things, <u>and</u>
  - c) achieve a good grade

- What do we expect you to learn to do?
  How much of an effort do you have to put in?
  How early do you have to be alert to deadlines?
    - Follow the rest of this intro and draw your own conclusions.

O1's Three Components

- The big primary component: assignments.

- End-of-chapter feedback forms:
  Submit a feedback from in each ebook
  chapter whose assignments you work on. (At
  least an estimate of the time you spent. Verbal
  feedback is voluntary but encouraged.)

- The third and final component is an
  end-of-course survey in December.

- The feedback you give does not affect your grade.
- The grade you get for assignments is your overall course grade.
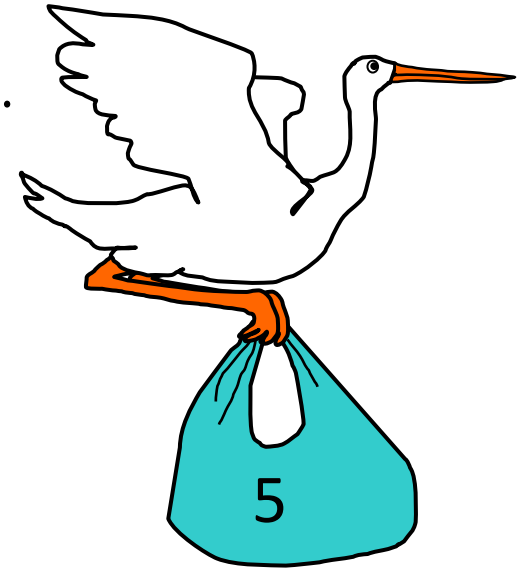
Assignments

- O1's assignments are split into numbered Weeks, from Week 1 to Week 12.

- Each Week has a deadline by which you need to submit that Week's assignments.

- The Week 1 deadline is on Wed, September 13th, at 18:00.

- The other deadlines are similarly on Wednesdays.

This lecture's most important point!

Assignment Categories

- The assignments are also split in Categories, which are labeled A, B, and C.
  - Category A: teach and assess skills that involve mandatory learning objectives → Grade 1
  - Category B: highly recommended, especially if you plan to take follow-on courses → Grades 2 and 3
  - Category C: For those who want to learn as much as possible; voluntary but useful → Grades 4 and 5

- See the first chapter of our course materials for more details.

There are also some bonus activities that don't affect your grade.

The assignments and accompanying materials are on our web site. It is on a course platform called A+, here:
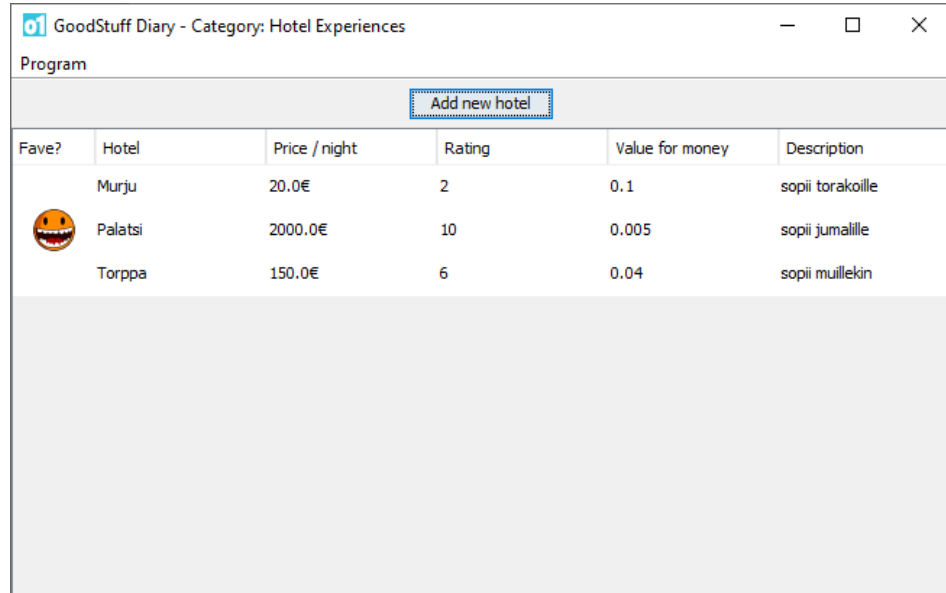
https://plus.cs.aalto.fi/o1/

(You'll find this link in MyCourses, too.)

What kinds of assignments are in store?

Week 1 (Category A)

- Make small changes to existing programs (an experience diary, a Pong game).

- Try instructing the computer with individual commands. E.g., arithmetic, playing sound, loading images from the net.

- Create small programs that, among other things: give a player a penalty to their score, convert units of measurement, animate a sequence of images, etc.
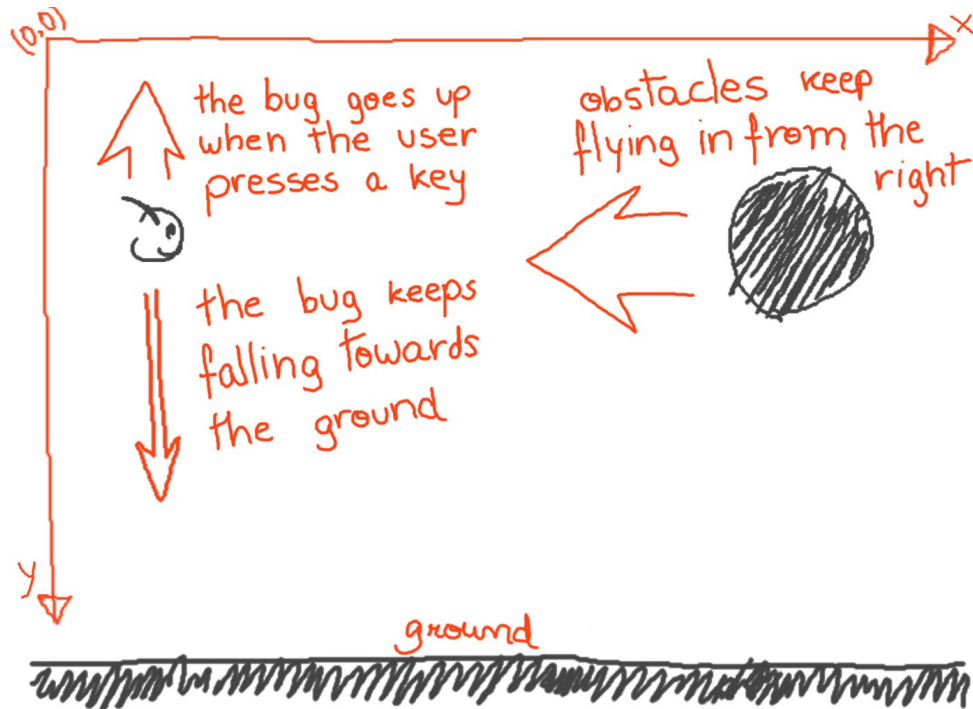
Week 2 (Category A)

- Implement a program component that represents "bank accounts" and records related events (deposits, withdrawals).

- Transform images via programming: rotate, scale, crop, "clownify", combine; solve an image puzzle.

This, like many of our assignments, involves a so-called toy example. Of course, we won't be dealing with real-world banking programs. CS1's examples often simplify things a lot, so that we can focus on learning the basics.

- Program an initial draft of a graphical game in which a "bug" tries to avoid obstacles.

(0,0) x

the bug goes up when the user presses a key

obstacles keep flying in from the right

the bug keeps falling towards the ground

y

ground

Week 3 (Categories A and B)

- Develop the bug game further: add moving graphics; make it less predictable by adding randomness.

- Create a program for tracking the results of football matches.

This is one of many programs in O1 that you don't write from scratch. Instead, you'll modify a given, incomplete program.

Week 4 (Categories A and B)

- Create a progam that constructs and draws maps of the night sky based on (real) star data.

- Implement new features in the football-scores program.

- Implement a car-on-a-map simulator.
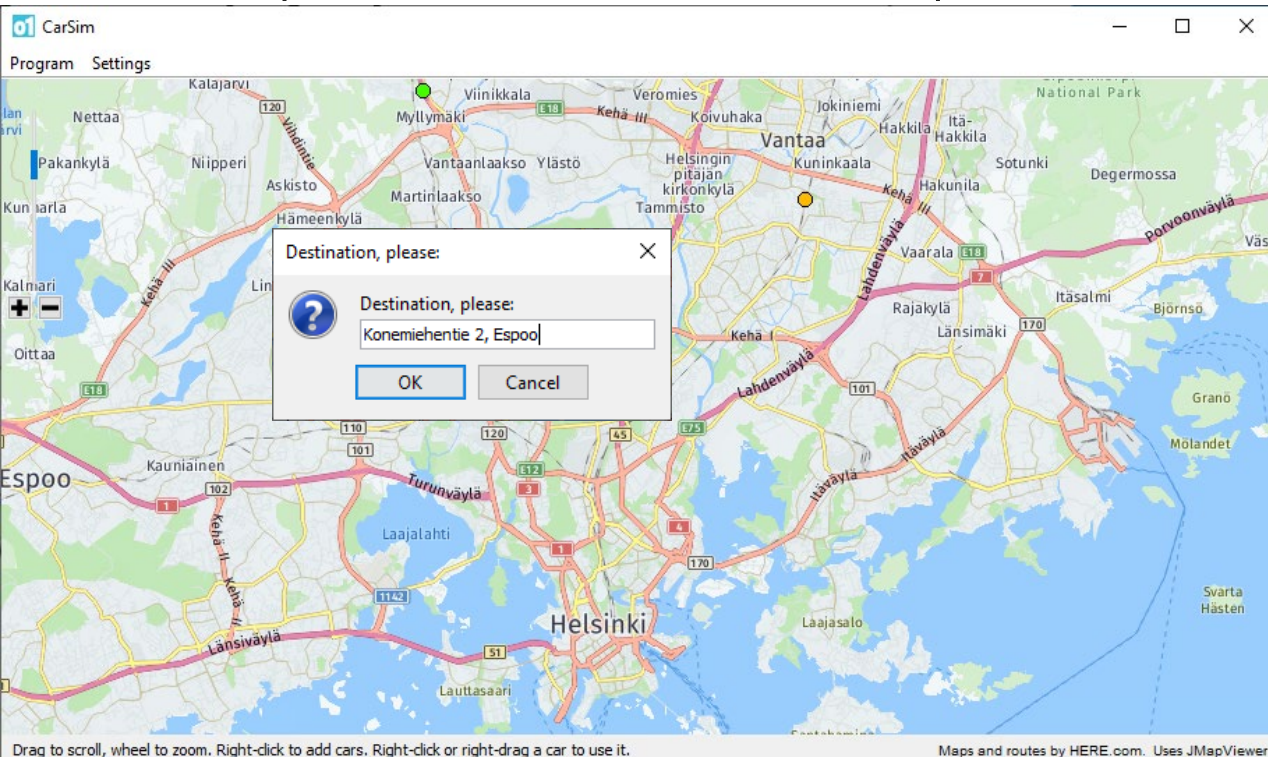
Week 5 (Categories A and B)

- Create programmatic models of various things, such as:
  - items for sale in an "online auction house" ,
  - DNA components,
  - the (in)compability of blood types.

- Create a program that computes election results for voting districts.
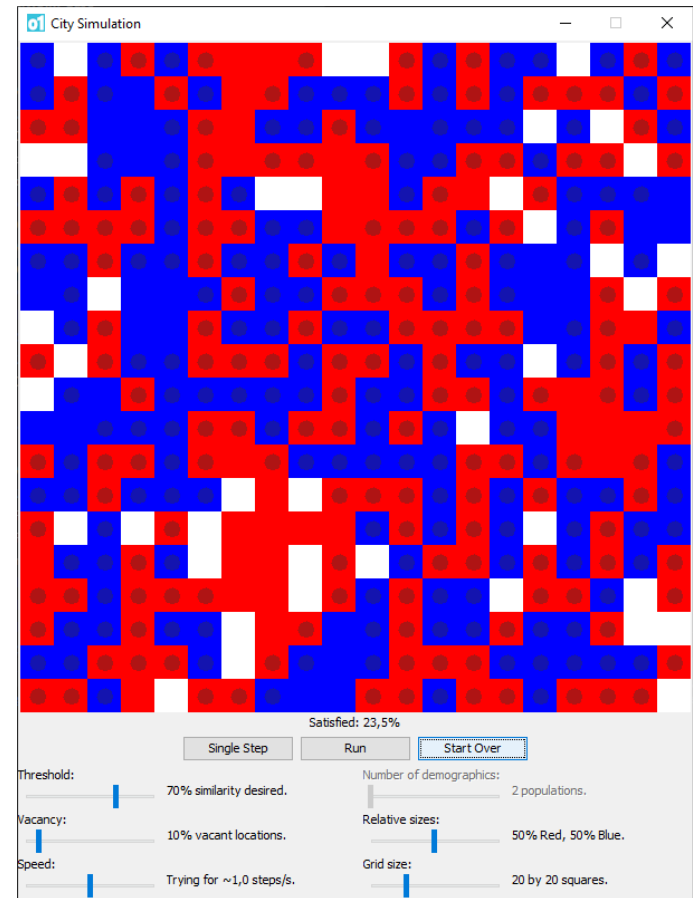
Week 6 (Categories A, B, and C)

- Create a program that determines which famous person's image is hidden in this messy-looking picture.

- Create a Snake game.
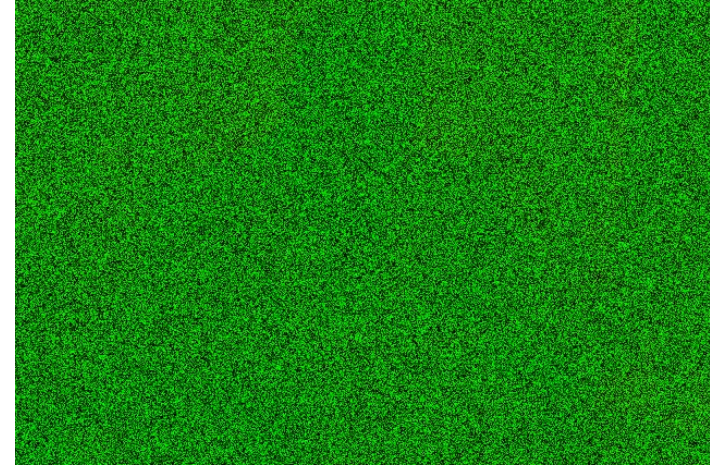
Week 7 (Categories A, B, and C)

- Create a program that computes statistics from measurement data.

- Modify a given program that "analyzes sentiments" from text.

- Model the house-moving behavior of residents on a city map.

Week 8 (Categories B and C)

- Crete a mine-sweeper-like game.
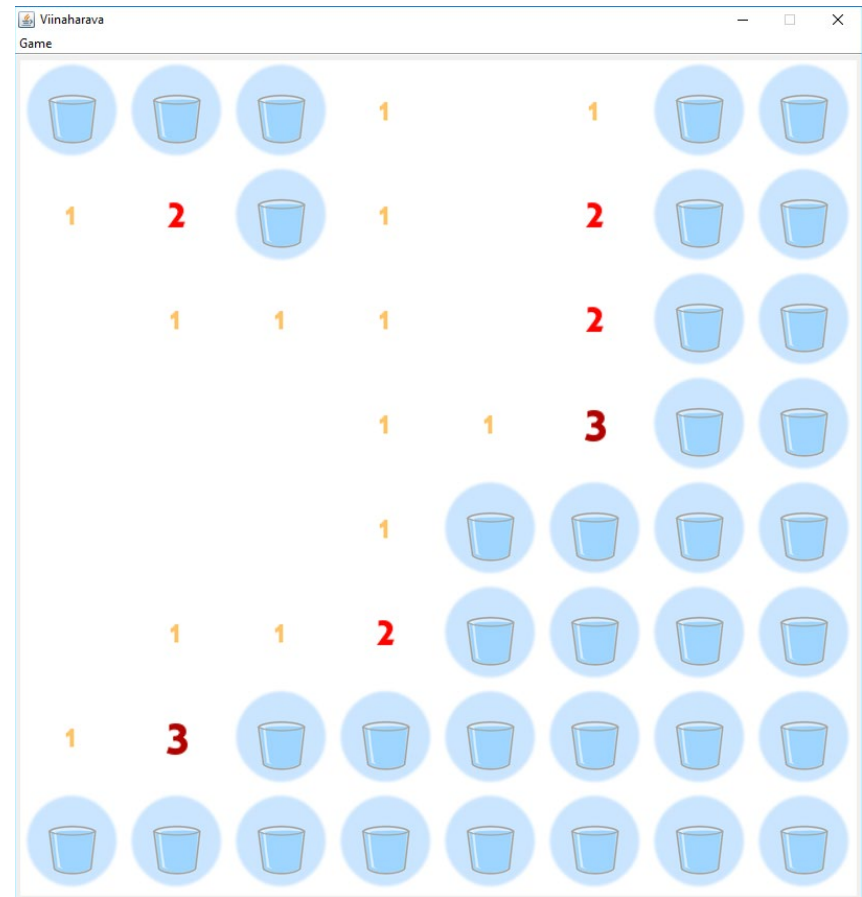
- Study and modify a given "robot simulator" program. This assignments continues into the next Week, where...

Week 9 (Categories B and C)

… you'll equip lovebots, slaybots, and others with different kinds of "brains".
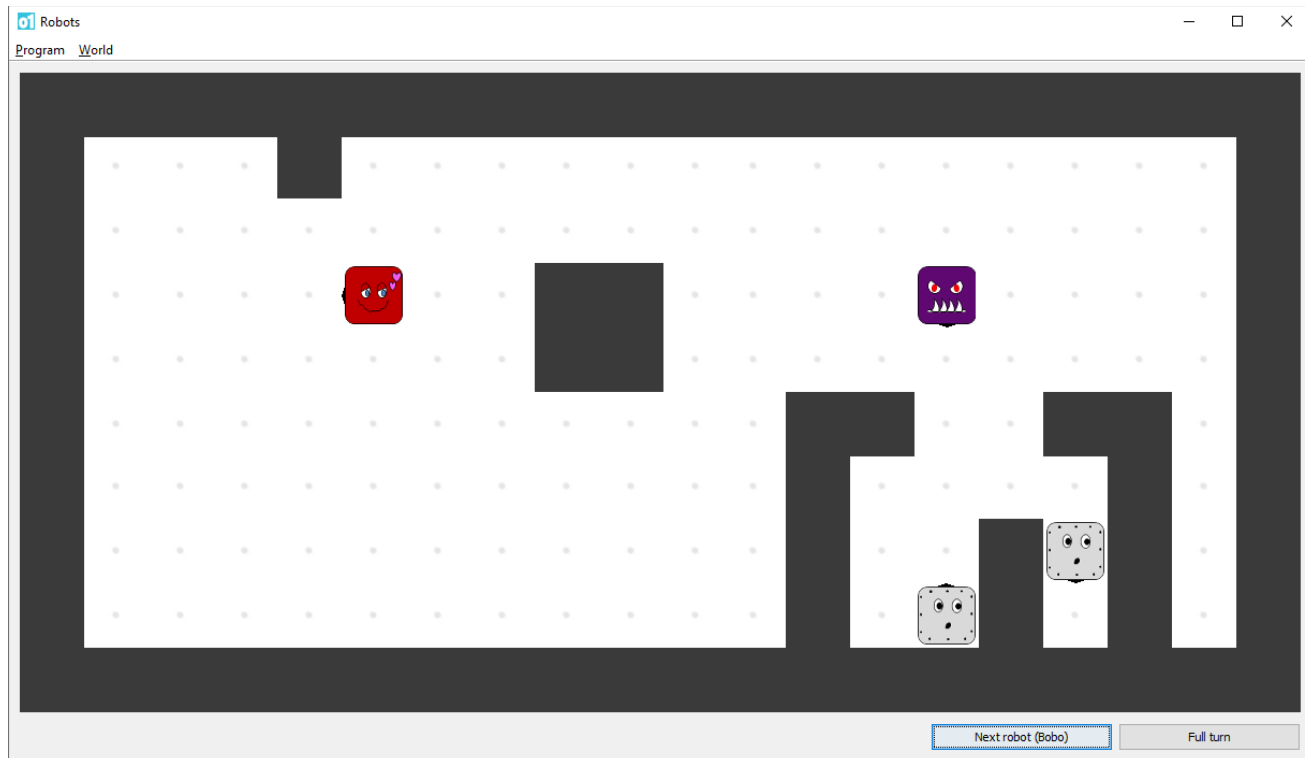
Week 10 (Categories B and C)

- PEEVELI!

A I _ A _ A N A

Arvattu: A, E, O, U, H, I, V, N
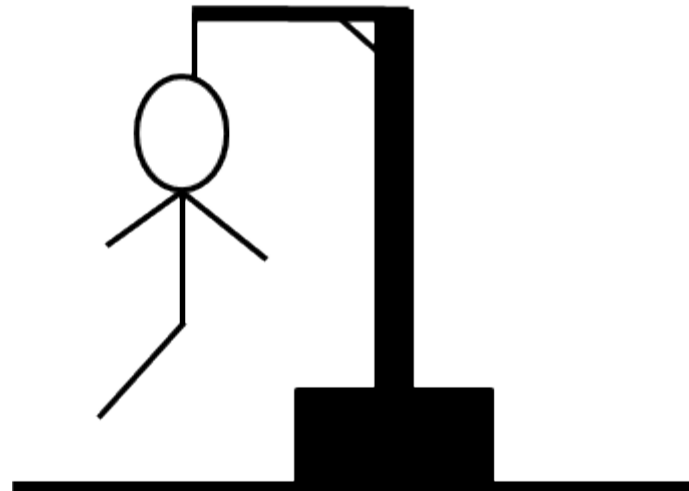
- Various follow-up assignments; continue with star maps and election results, for example.

Week 11 (Categories B and C)

Make whichever sort of text-based adventure game you like! Creative solutions welcome!

(We'll give you a base program that you can edit as
you like or even ignore completely if you prefer.)

Week 12 (Categories B and C)

- Fix problems in a given train-ticketing program

- Study recursion as a programming technique.

- Various optional materials, including a "robot tournament" between students.
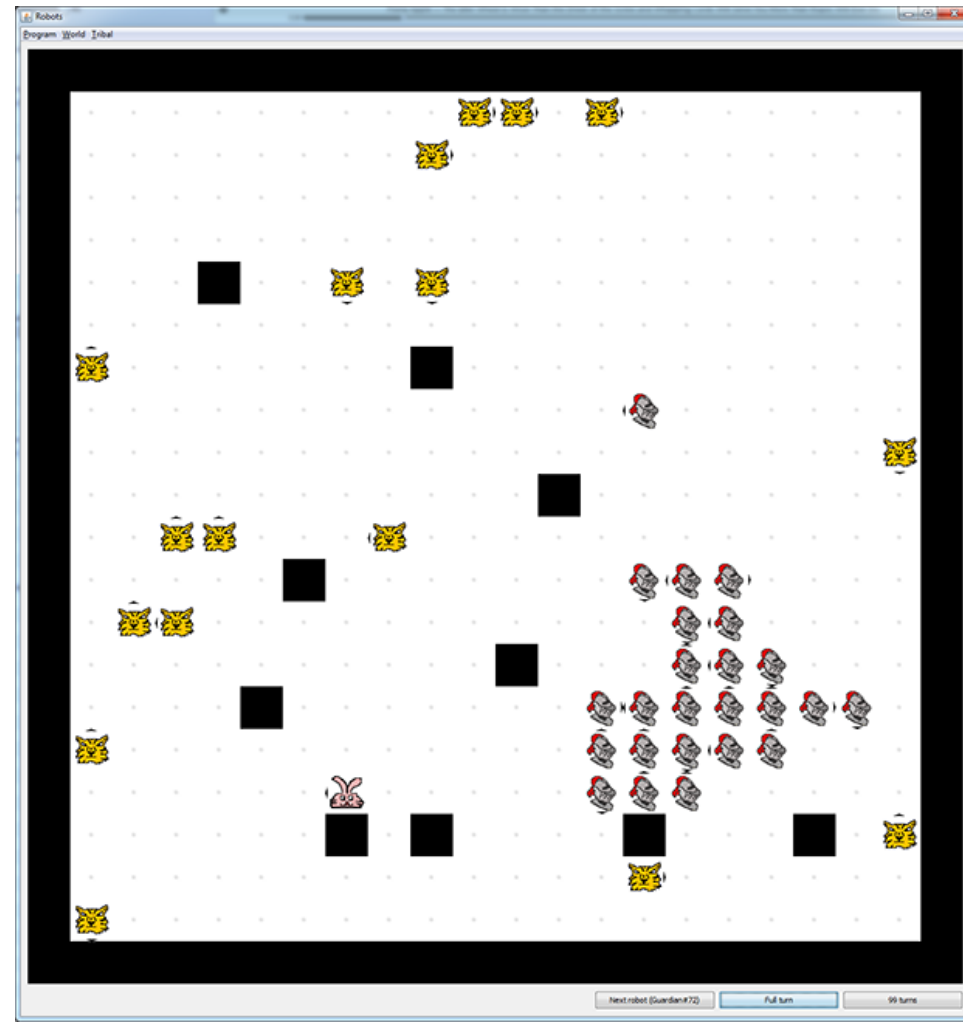
Week 13

- Just our end-of-course feedback survey, no regular assignments.

The Moral of the Story

- The majority of O1's assignments are practical: you need to make things.

- You can't solve most of them by just memorizing some basic facts or trying to copy an obvious answer from a textbook.

  Fortunately, most students get excited enough about programming that this isn't much a problem. ;-)

- Doing these assignments is a crucial and mandatory component of O1.

This is Boring But Needs to be Said

- It is forbidden to use "artificial intelligence" (AI) to solve O1's assignments.

- In other words, you are not allowed to use technical tools that generate program code or other solutions to assignments (e.g., ChatGPT)
  - Such use counts as plagiarism just like copying solutions from a human does.
  - Each student or student pair must submit only solutions that they themselves produced.

- You do not have permission to enter our course materials into AI tools or to otherwise distribute the materials to external parties.

Working to Build Skills

- To know programming means not only to have knowledge but to have skills.

- You cannot to build up a skill merely by reading or listening.

- Learning a skill tends to take time...

Train Your Programming Brain

- Cognitive science tells us:
  - Brains are rather like muscles in that they work better if you train them.

- And such brain-training is rather topic-specific.
  - A complex skill demands lots of practice.
  - To become really good at something takes years of practice in that specific area.
- A high IQ won't save you from needing to practice.

- In O1, you'll need to train your "programming brain".

O1 as a Workout Plan

- For you to learn (and to do well in O1), your need to make a personal effort!

- The course is your "workout plan".

- We in the O1 staff can help as your "personal trainers", but we can't exercise for you.

Course Materials

- The A+ course platform is your "gym".

- In A+, you'll find our ebook: a combination of assignments and textbook-like interactive learning materials.

- You complete O1 by working through that ebook and solving the assignments that are embedded there.
  - There's plenty of reading and plenty of doing. (The early Weeks involve more reading than the later ones do.)

Five Credits

- 5 ECTS credits ≈ 130 *active* hours of study for a "typical student".
  - That's a pretty big number.
  - If you go through all our optional materials (and aim for a grade of 5), O1 may be still more work.

- That can easily mean more than 10 hours in some Weeks.

Don't leave everything for deadline day!

You Don't Need to Do It Alone

- Pairwork

- Lab sessions

- Online forums

- Weekly bulletins

Make use of these!
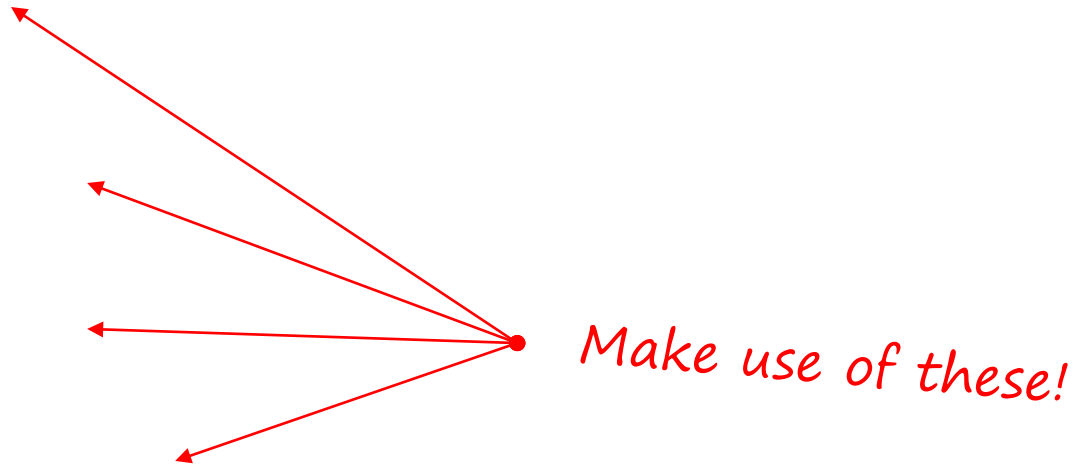
Pairwork is Allowed and Encouraged!

- You might want to form a pair right away in Week 1.
- You can use our online channels to assist you with that.
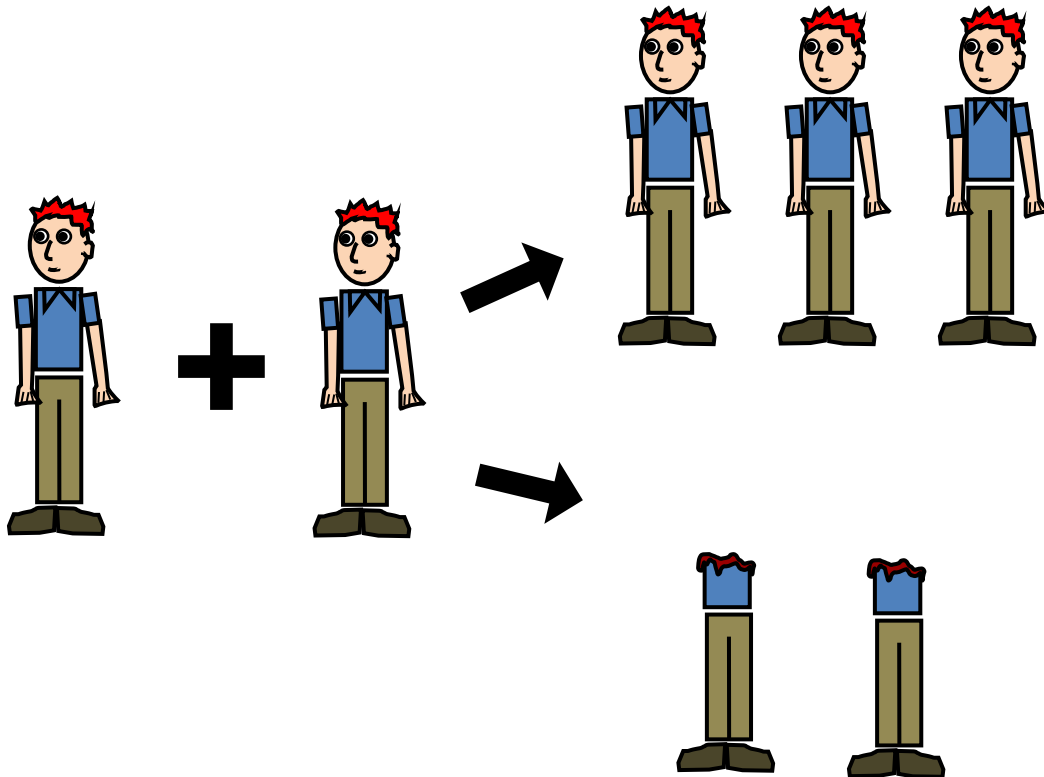
- It's also allowed to solve the assignments alone.

Lab Sessions

- You're most welcome to join our lab sessions, where you can work on the assignments at your own pace and ask for advice as needed.

- There are over 40 hours of labs per week; there are some every day, except on weekends.

- You'll find the times and places in A+.
  – There are a few online sessions, but the labs are mostly on campus.

- No need to pregister. Just come and go freely.

Piazza and Telegram

- We have a questions-and-answers forum online, on the Piazza platform.
    - You can submit questions for course staff and other students to answer.

- We have a group in Telegram.
    - This enables a real-time chat about O1.

- You'll find the links in A+'s menu and in the first chapter of our ebook.

Weekly Bulletins

- Published by Juha in A+ after each deadline.

- The bulletins complement our other materials with
  - answers to student questions and feedback
  - example solutions to past assignments, and
  - various bonus topics.

Lectures

- After this intro lecture, we'll only have one, and that isn't exactly a traditional lecture:

The time is tentative and might still end up being changed!

- End-of-Course event on Friday, December 8th, 13–15.
  - We'll look at what we (you!) have achieved, and consider possible next steps after O1.
  - We'll take a look at the teaching assistants' favorite text adventure games, give out some awards for those, run the finals of the robot tournals, etc.

Welcome!

- Despite all the warnings about workload, O1 is perfectly doable, as long as you're active!

- We've tried to make everything as student-friendly as possible, so that your working hours are well spent and effective for learning.
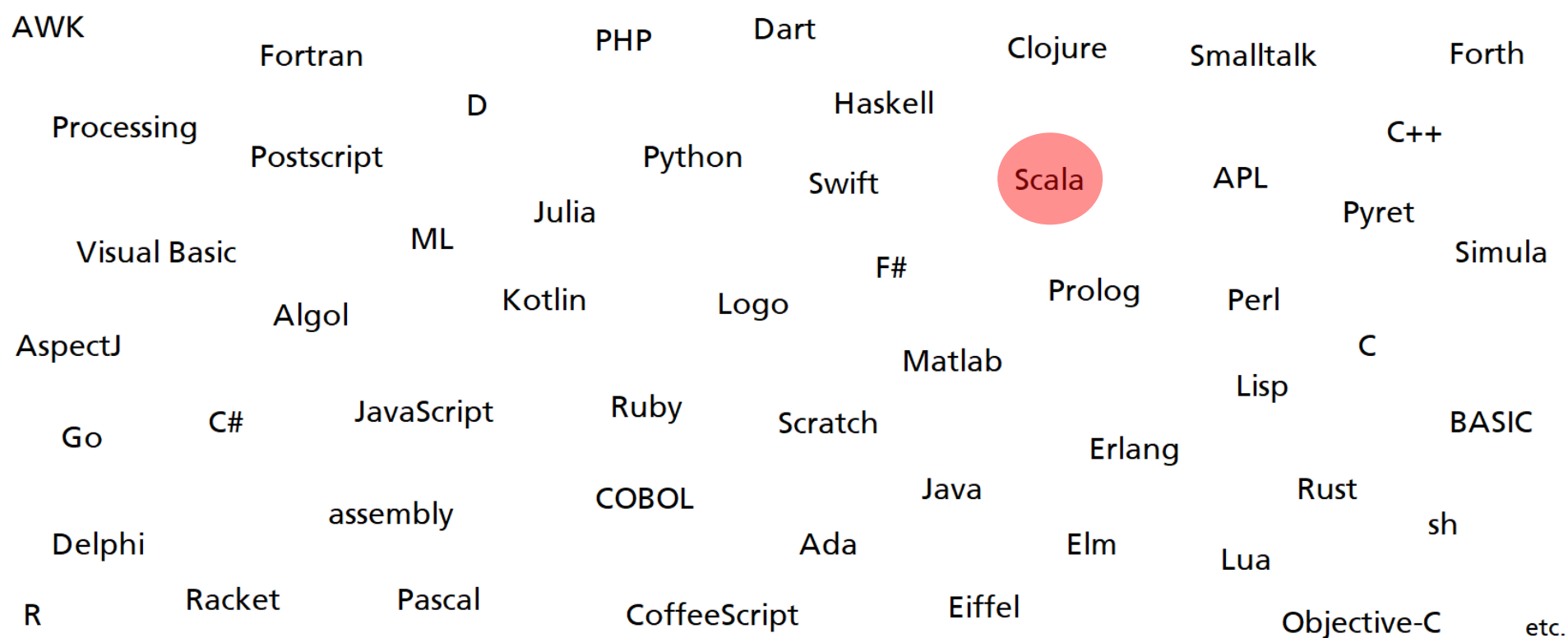
Now, let's get to programming…

Programming?

- To program is to create instructions for a computer to follow.
  - When a computer runs a program, things happen as specified by the programmer.

- An *application* is a common sort of program.
  - Each application has a specific purpose. It helps people do meaningful things related to that purpose.
 - Examples: an email program, a text editor, a computer game.

Programming Languages

- To instruct a computer, we need a suitable language, a *programming language*.

- There are loads of options.

- In O1, we'll be using a particular language. At the same time — and more importantly — you'll learn general programming skills and principles that apply generally to many languages.

AWK
Fortran
PHP
Dart
Clojure
Smalltalk
Forth
D
Haskell
Processing
C++
Postscript
Python
Scala
APL
Swift
Pyret
Julia
Visual Basic
ML
Simula
F#
AspectJ
Algol
Kotlin
Logo
Prolog
Perl
C
Matlab
Lisp
Go
C#
JavaScript
Ruby
Scratch
BASIC
Erlang
COBOL
Java
Rust
assembly
sh
Delphi
Ada
Elm
Lua
R
Racket
Pascal
CoffeeScript
Eiffel
Objective-C
etc.

IntelliJ, a Programming Environment

- Programmers use existing programs as tools that help them create new ones.

- There are loads of such helper programs to choose from.

- In O1, we'll be using a program called *IntelliJ IDEA.*
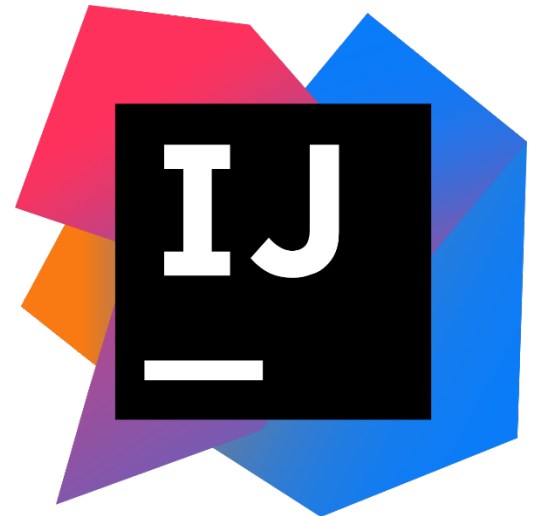
Things to Note Right Now

- *The first deadline is on Wednesday, September 13th, at 18:00.*
  - Before that, you'll need to study prenty of material in Week 1 and submit the associated assignments.
  - Get started soon!

- We'll have lab sessions every day (weekends excluded), starting from this week's Thursday.
  - Details in A+.

- See you at the lab sessions, on our online channels, and in December's end-of-course event!