# Computational Macroeconomics

## Lecture 1: Introduction

Kanato Nakakuni

October 20, 2025

# Course objectives

1. Introduce computational methods, techniques, and procedures for solving and analyzing macroeconomic models

2. Develop practical skills in computationally solving macroeconomic models through hands-on exercises

We will achieve these goals by working mainly on:

- **Overlapping generations (OLG)/Lifecycle model**
    - One of the core frameworks in macroeconomics
    - Especially useful when interested in topics related to demographics (population aging, social security) and household behavior over the lifecycle (human capital accumulation, fertility, marriage/divorce, retirement, education investment, etc.)
- Its extension incorporating household **heterogeneity** (income, family structure, etc.)

# Course structure

- Part 1: Solving household dynamic optimization problems (Lectures 2 and 3)
    - Keyword: discretization, grid search, endogenous grid methods, backward induction
- Part 2: Solving for equilibrium (Lectures 4–7)
    - Keyword: stationary equilibrium, transitional dynamics, calibration, counterfactual analysis
    - Application to macro implications of social security reforms
- Part 3: Notes on other important methods (Lectures 8–10)
    - Keyword: value (policy) function iteration, root finding, optimization, interpolation
- Part 4: Advanced topics and application (Lectures 11–14)
    - Keyword: Aiyagari model, discretizing stochastic income process, heterogeneity
    - Application to macro effects of family policy

# Basic information

- **Period**: October 20, 2025 – December 5, 2025 (+ final exam)
- **Time**: Monday 13:45–15:15 and Friday 8:30–10:00
- **Place**: 158 PC pool (L7, 3-5)
- **Prerequisite**: Intermediate-level courses in micro and macro
- **Grading**: Assignments (50%) + Final exam (50%, 90 min)
    - Tentative plan: A coding assignment will be posted in the first week of November, with a deadline of December 5, 2025.

# Basic information (cont'd)

- **Instructor**: Kanato Nakakuni (kanato.nakakuni@uni-mannheim.de)
- **Teaching assistant**: Mingjie Xu (mixu@mail.uni-mannheim.de) will lead couple of coding sessions during regular class time (no separate TA sessions)
- **Course materials**: Provided on my GitHub repository, *ComputationalMacro*
  - Please send me an email today with your name and email address. In reply, I will send access invitation to the GitHub repository

# Programing language

- In explaining sample code, I will use Julia
- You may use other programming language (e.g., MATLAB, Fortran, Python) for in-class exercises and to submit your coding assignments if that is more comfortable
- Otherwise, please install Julia and a suitable code editor (e.g., Visual Studio Code) on your own laptop
- Please install them before the next class. Links for installation:
  - Julia: https://julialang.org/install/
  - VS code: https://code.visualstudio.com/