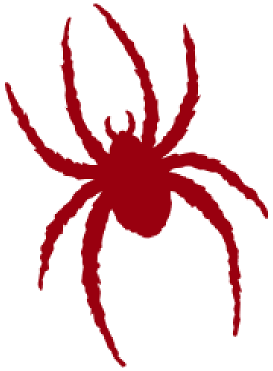
UNIVERSITY OF RICHMOND

Welcome to CMSC 326!

CMSC 326 Simulation

Today

- Introductions
- Course logistics
- Motivation
- What is a Simulation?
- Environment setup
- In-class coding exercise



Introductions

Dr. David Balash

Faculty page: https://cs.richmond.edu/faculty/dbalash

Homepage: https://davidbalash.github.io

Professor Balash

“Ba-lish”

He/Him

- BS in computer engineering Iowa State
- Two-decade career as a software engineer
- MS and PhD in computer science from GW
- Research: Computer S&P

Dr. David Balash

Things I like

- Education/Learning
- Hiking
- Cycling
- Guitars
- Board games
- Programming
- Cats

Ask me anything

RICHMOND

Assignment 1

Task: Create a personal introduction slide and post it to the introductions channel on the course Slack workspace

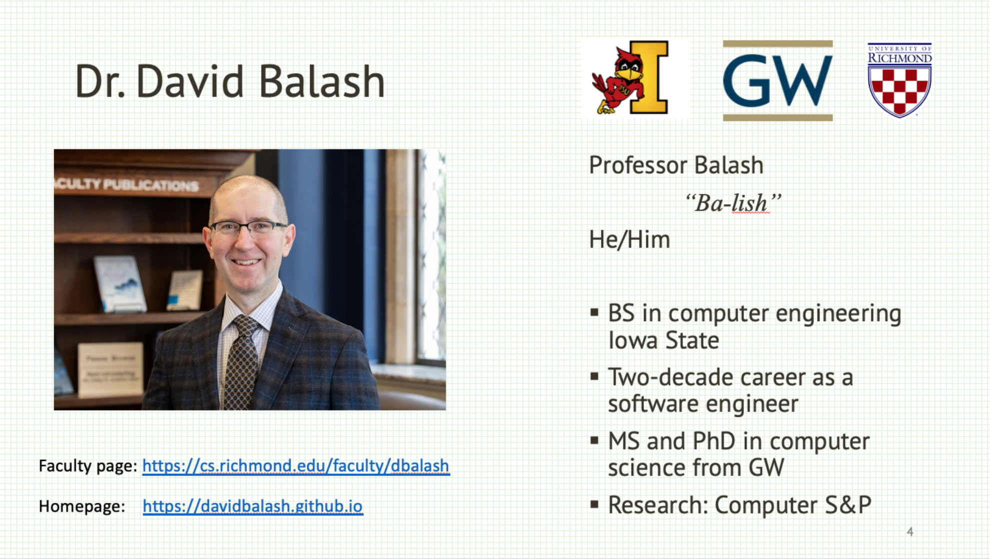
Due: Friday by 11:59 PM

Points: 5

Be Creative

Name

Photo



Dr. David Balash

Professor Balash

“Ba-lish”

He/Him

- BS in computer engineering Iowa State
- Two-decade career as a software engineer
- MS and PhD in computer science from GW
- Research: Computer S&P

Faculty page: https://cs.richmond.edu/faculty/dbalash

Homepage: https://davidbalash.github.io

Pronunciation

Pronouns

Personal Introduction

Classroom Meet and Greet

1. Introduce yourself to a person near you

2. Introduce yourself to a different person near you

- Potential conversation topics:
  - What are some of the things that you like?
  - Who are your favorite pets?
  - Why do you want to take this class?

Student Introductions

- Name
- Pronouns (optional)
- Major
- Class year
- Favorite snack food

Course Logistics

Classroom Norms

- Questions are always welcome!!
  - Ask them at any time
- “I don’t know” is okay
- Be curious
- Treat peers and instructors with kindness and respect
- Communication is key!
- Seek support when needed

Where All Class Information Can Be Found

https://cmsc326-s25.github.io

How to Communicate With Me

- Slack workspace
  - https://cmsc326-s25.slack.com

- After class or in office hours - 223 Jepson Hall
  - Tue 4:30PM - 6:00PM
  - Thr 4:30PM - 6:00PM
  - and by appointment
- Email
  - david.balash@richmond.edu

Topics Covered

- Computational mathematics
- Simulating continuous systems
- Discrete-Event simulation
- Probability and simulation

Learning Outcomes

- Understand the key concepts of simulation, including discrete-event simulation, Monte Carlo simulation, and stochastic modeling
- Develop discrete-event simulations for systems involving queuing and inventory management
- Develop Monte Carlo simulations to estimate probabilities, model random processes
- Apply elementary statistics to analyze simulation outputs
- Write simulation programs using the Python programming language and libraries

Lecture

- Will usually include in-class module exercises
- In-class exercises will be due one or two weeks from when they are assigned (except during break)
- Regular attendance is expected
- Students who are sick should not attend class
- Notify me in advance of the absence, if possible

Labs

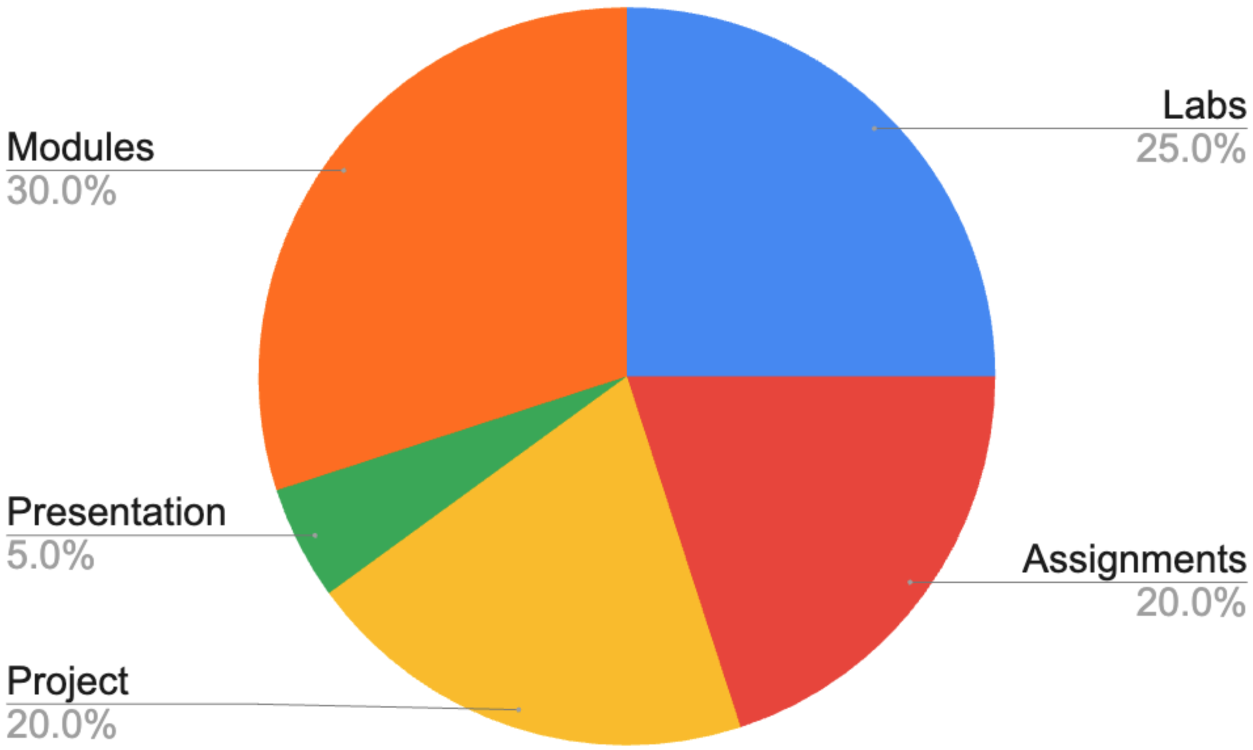
- Lab assignments done individually and in groups
  - but will always be turned in individually

- Lab assignments are typically due at 11:59 pm on the night prior to the next lab (except during break)

- Please ask for help when needed

Coursework and Grading

- Modules (In-class coding exercises)
- Lab assignments
- Programming project
- Project Presentation
- Programming Assignments



Textbook

- No textbook

- Reading materials may be assigned during the semester

Ask me a question

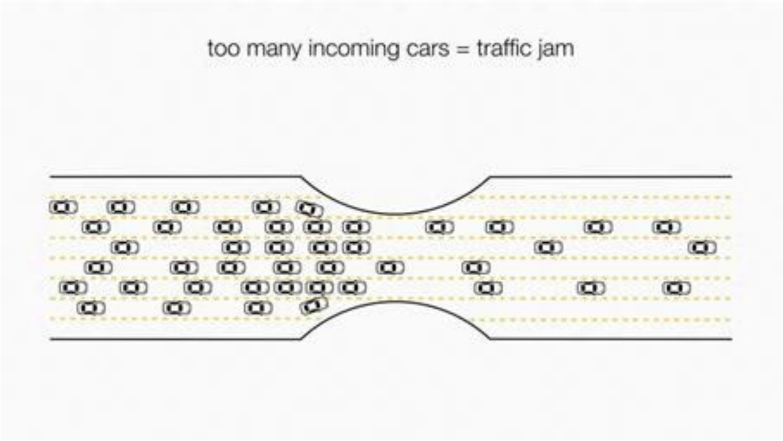
Motivation

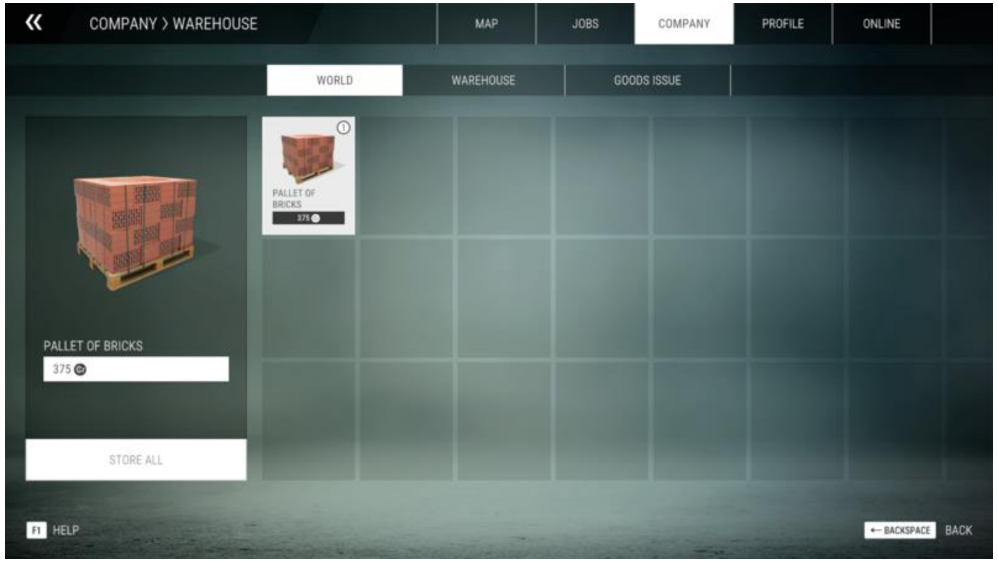
Why Take a Simulation Course?

What if you could predict the future… using code?

Why Take a Simulation Course?

What if you could predict the future… using code?





Simulations are Everywhere

- Predict customer wait times
- Simulate inventory needs for businesses
- Analyze sports strategies and improve performance
- Optimize traffic flows in cities
- Model disease spread and evaluate public health policies

Why Learn Simulation Techniques?

- Solve problems in fields like healthcare, logistics, finance, and engineering
- Analyze uncertainty and randomness in real-world systems
- Test scenarios and “what-ifs” without real-world risks or costs
- Develop skills in programming, statistics, and critical thinking

Cool Applications of Simulation

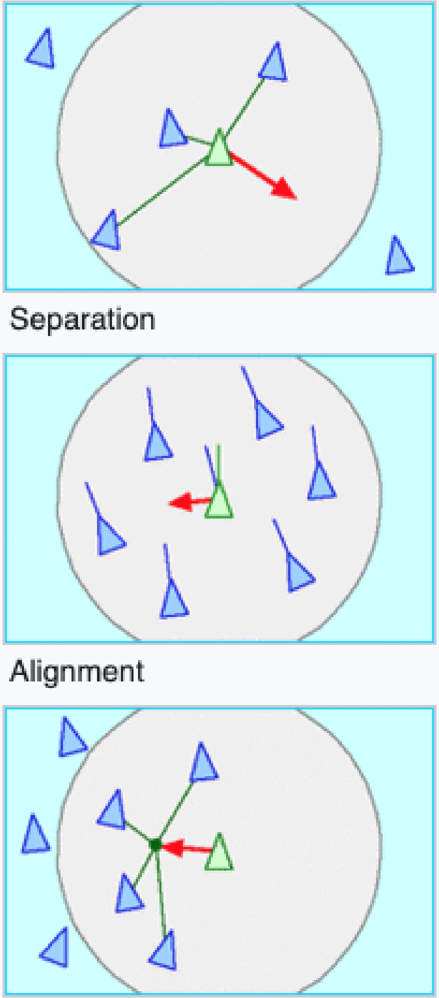
Cool Applications of Simulation

- **Boids** is an artificial life program, developed by Craig Reynolds in 1986, which simulates the flocking behavior of birds, and related group motion

- **separation**: steer to avoid crowding local flockmates

- **alignment**: steer towards the average heading of local flockmates

- **cohesion**: steer to move towards the average position (center of mass) of local flockmates



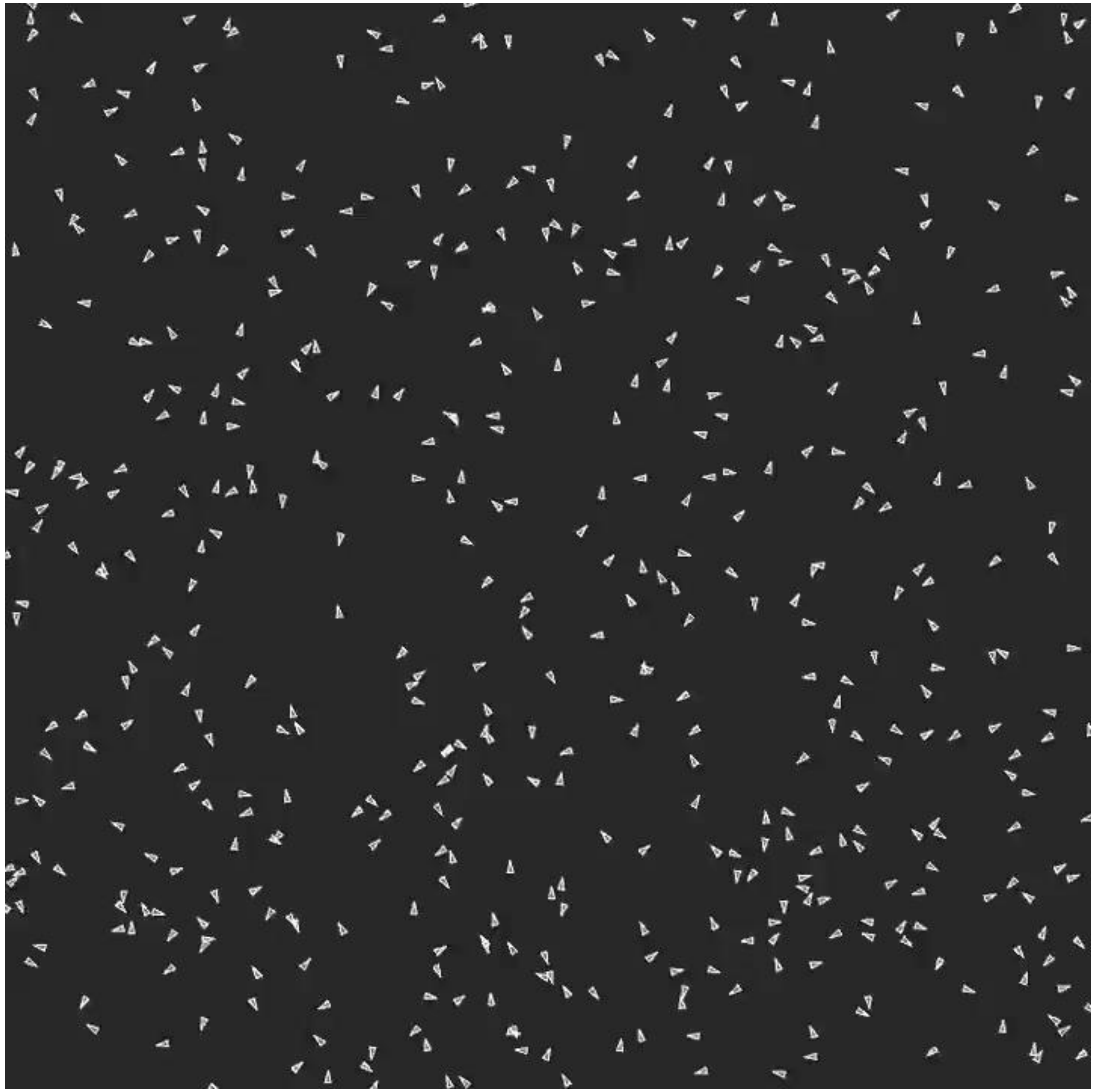
Rules applied in simple Boids

Separation

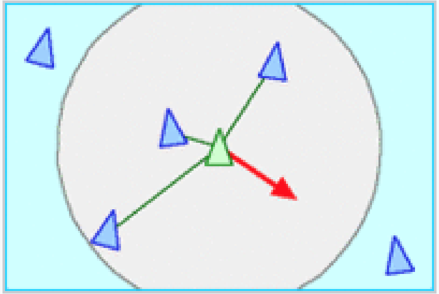
Alignment

Cohesion

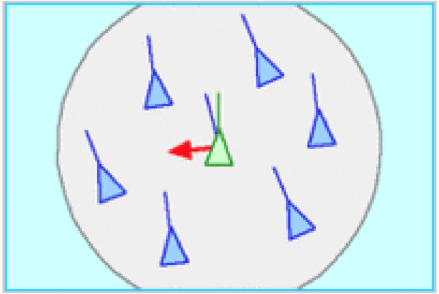
Cool Applications of Simulation



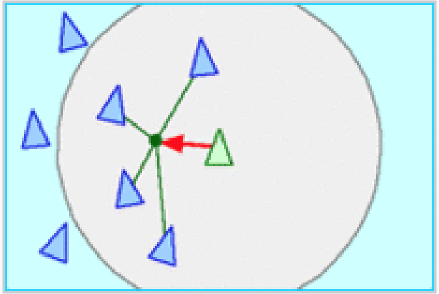
Rules applied in simple Boids



Separation



Alignment



Cohesion

Cool Applications of Simulation

- https://www.complexity-explorables.org/slides/berlin-8-am/

- https://www.complexity-explorables.org/slides/i-herd-you/

- https://www.complexity-explorables.org/slides/flockn-roll/

What is a Simulation?

Simulation

A **simulation** is the imitation of the operation of a real-world system over time.

- It involves:
  - The generation of an artificial history of a system
  - The observation of the history to draw inferences on the characteristics of the system

Simulation Model

The behavior of a system (as it evolves over time) can be observed through a **simulation model**.

- Models help to investigate a variety of “what if” questions about real-world systems
  - What would be the impact of changes on system performance?

- Many real-world systems are so complex
  - Can be solved with computer-based simulation models

System

A **system** is a group of **objects** that are joined together in regular interaction to accomplish a **purpose**.

- For example:
  - **System**: Automobile manufacturing system
  - **Objects**: Machines, component parts, workers
  - **Purpose**: Production of vehicles

Components of a System

The following are the components of a system (Example: Bank System)

- **Entity**: Objects of interest in the system
    - Clients
- **Attribute**: Properties of an entity
    - Balance in a client’s savings account
    - Credit rating
    - Account number
- **Activity**: A time period of specified length
    - Making deposits
    - Withdrawing cash

Components of a System

The following are the components of a system
(Example: Bank System)

- **State Variable**: Collection of variables necessary to describe the system at any time
  - Number of busy tellers
- **Event**: An instantaneous occurrence that might change the state of the system
  - Arrival of a client
  - Service completion of a client

Components of a System

Example: Rapid Rail System

- **Entities**: Riders
- **Attributes**: Origin, destination
- **Activities**: Traveling
- **State Variables**: Number of riders at each station, number of riders in transit
- **Events**: Arrival at a station, arrival at a destination

Components of a System

Example: Global Inventory System

- **Entities**: Warehouses
- **Attributes**: Capacity
- **Activities**: Withdrawing
- **State Variables**: Level of inventory, backlogged demand, rejected demand
- **Events**: Demand

Discrete and Continuous Systems

A system is **discrete** if the state variables are changing at a discrete set of time points

- bank, multi-level car park, board game, assembly line

A system is **continuous** if the state variables are changing at a continuous set of time points

- water dam, chemical reaction, ball rolling down a hill

The Need For a Model

We can study a system through direct experimentation on the system itself. However, this is often not an option:

- system might not exist yet
- it may be impractical to experiment with the system

Most studies require a model of the system.

Definition of Model

A **model** is a representation of a system for the purpose of studying the system.

- should consider only the aspects of the real system that affect the problem under investigation
- should be sufficiently detailed to permit valid conclusions

A model of the system has the same components as the real system (entities, attributes, etc.), **but only the ones that are relevant to the problem**

Types of Models

- A **physical model** is a larger or smaller version of a system
- A **mathematical model** uses symbolic notation and mathematical equations and inequalities to represent a system
  - A **simulation model** is a particular type of mathematical model of a system (usually in software)

Types of Simulation Models

- Static vs. Dynamic
  - **Static simulation models** represent systems at a particular point of time
    - the total newspapers sold at the end of a specific day
  - **Dynamic simulation models** represent systems as they change over time
    - the progress of newspaper sales between 9:00 AM and 4:00 PM

Types of Simulation Models

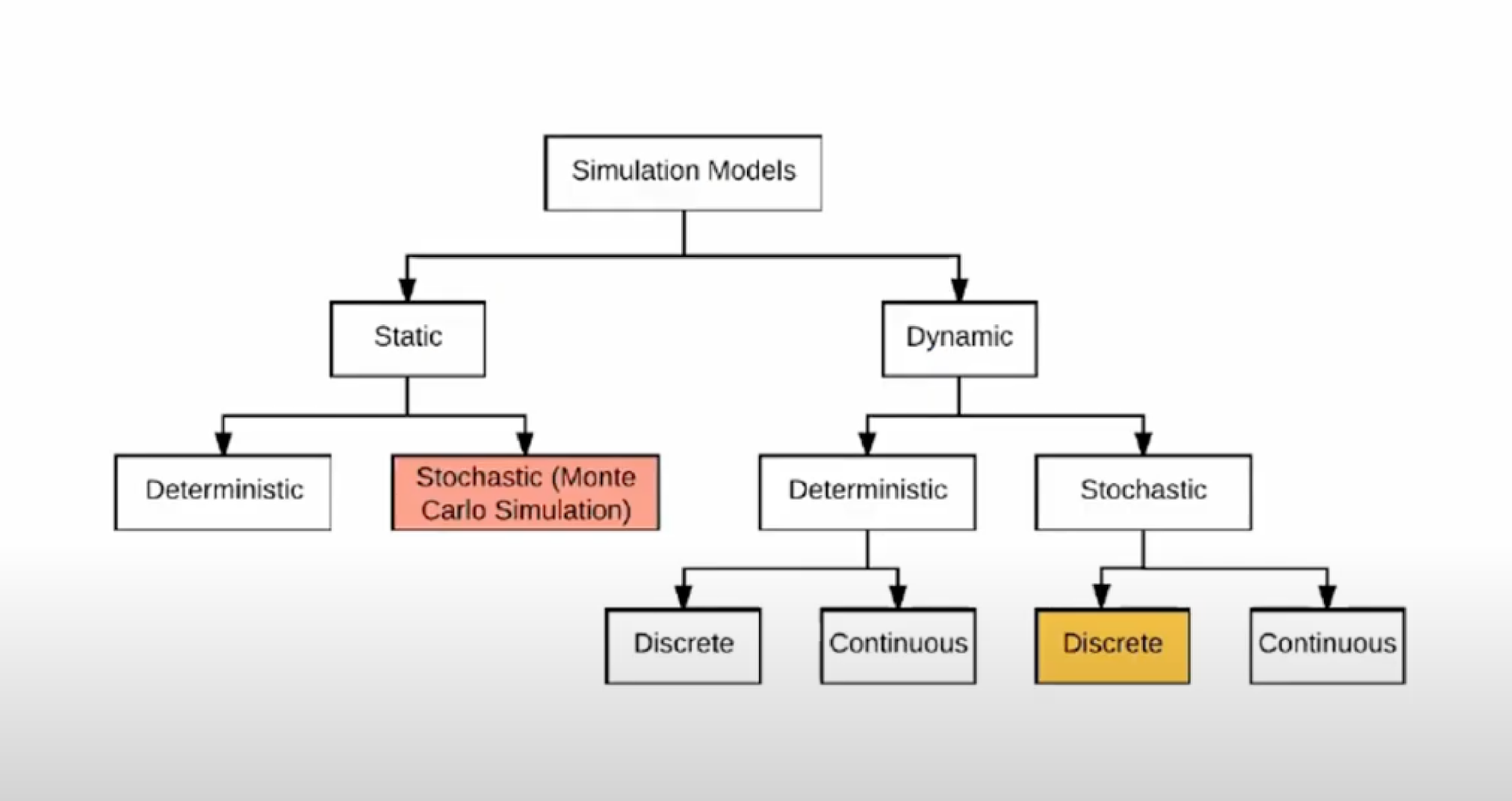
- Deterministic vs. Stochastic
  - **Deterministic simulation models** have known inputs (variables)
    - the arrivals which are based on scheduled appointment times are deterministic
  - **Stochastic simulation models** have random (probabilistic) inputs (variables)
    - random arrival and service times are stochastic

- If there is, at least, one random input in the model, then the model is stochastic

Types of Simulation Models

- Discrete vs. Continuous
  - Just like the systems are discrete or continuous, the models are either discrete or continuous
  - If there is, at least, one state variable that changes at a continuous set of time points, then the model is continuous

Types of Simulation Models



Simulation Models

Static

Dynamic

Deterministic

Stochastic (Monte Carlo Simulation)

Deterministic

Stochastic

Discrete

Continuous

Discrete

Continuous

Environment Setup

In-Class Coding Exercise