PERFORMANCE ENGINEERING

Lecture 0: Introduction

April 4th , 2022

Teacher: Ana Lucia Varbanescu
a.l.varbanescu@uva.nl
Super-TA/Advisor: Stephen Swatman
s.n.swatman@uva.nl

About me …

- BSc and MSc in Computer Science and Engineering from POLITEHNICA University of Bucharest, Romania
- PhD from TUDelft, NL
  - Thesis in performance engineering
- Postdoc till 2013 at VU University and TUDelft
- At UvA since June 2013 **till December 31st, 2021**

- 15+ years experience in performance engineering
  - Focus on parallel systems and GPUs
- Supervisor of various projects in performance engineering
  - BSc, MSc, PhD

Teaching principles [1]

- Lectures are meant to discuss the challenging topics
  - Not all slides are covered
- Discussion = dialogue
  - I ask a lot of questions => keep you alert
  - Please try to answer => easier to learn
  - No trick questions => not always easy answers…
- Your questions are welcome
  - Interaction is the most important activity during class

Teaching principles [2]

Quiz = a 5-15 minutes activity in which you should answer a few short questions on-the-spot.

- Bring pen and paper.
- Can use online forms… preference?

- I like quizzes!
  - At least 1 per lecture

- I grade quizzes
  - Discuss answers in the following lecture
  - Quizzes score => /70 added to the final grade
    - E.g.: 50 points => 0.7 bonus to the final grade.

Expectations

During class:

- Ask a lot of questions
  - And answer my questions
- Try to answer quizzes
  - And if things are not clear, let me know

- Presence in lectures: voluntary, recommended
  - Most slides will be online.
    - Including lecture notes
  - Additional reading will be posted online.

- No recordings of ANY KIND without permission *for every instance*

Course structure

- Lectures : 9-10
  - Theoretical and empirical concepts
  - Guidelines
  - Further reading material
- Labs: 7
  - 4 Small-scale assignments
    - Limited coding effort
    - Focus on performance analysis, modeling, and prediction
- Seminars: 4-5
  - Dedicated to the project (next next slide)
- Presentation sessions: 2
  - Dedicated to the project

Grading

- 20% exam
  - From the lectures
  - Intended to test your basic knowledge on performance engineering
- 30% assignments
  - Grade for the code and report
  - Focus on the performance engineering goals
- 50% project
  - Work + report = 35%
  - Presentations = 15%
- Bonus: from quizzes and assignments.

Project [1]

- Idea: a coherent performance engineering project/effort
  - Start: an application + performance requirements
  - Optional constraints: an architecture
  - Goal: meet the performance requirements through PE
- Components
  - Implementation
  - Analysis
    - Tools, methods, …
  - Reporting

Project [2]

- Kick-off … on Thursday, April 7th
  - More details on structure, expectations, rules, etc.
- Mid-term presentation – May 12th
- Exam & Final presentation – June 2nd (TBD)
- Various invited speakers in the meantime …

Learning objectives

"At the end of this course, students will be able to...."

- LO1: **quantify** (using the appropriate tools and methods) the **performance** of an application running on a computing system using the appropriate **metric**.
- LO2: demonstrate and compare several **performance modeling methods**, and assess their usefulness for practical problems.
- LO3: classify and use several **performance prediction methods**, and compare their applicability in practice.
- LO4: **design an empirical performance analysis** process for any application, interpret its results, and **recommend solutions for performance improvement**.
- LO5: design and **use a suitable model for accurate performance prediction** for a given application.
- LO6: **design** and develop a **complete performance engineering process**, apply it successfully on any given application, and assess its outcome in terms of performance gain.
- LO7: Use different **performance engineering tools** (e.g., profilers, microbenchmarks and benchmarks, performance counters libraries, etc.).

Table of contents (WiP)

- Basics of performance, code tuning, basic models
- The Roofline model and extensions
- Analytical modeling
- Benchmarking & microbenchmarking
- Data-driven and statistical modeling
- Simulation and simulators
- Performance counters and performance patterns
- Lost and found topics: queuing theory, the polyhedral model, new developments