

Distributed and Parallel Programming

Bachelor course 5062COPP6Y, 2025/2026 Sem. 1, Per. 2



Lecturers: dr. Zhiming Zhao¹ (*coordinator*), dr. Dolly Sapra² and dr. Adam Belloum^{1, 3}

¹Multiscale Networked System (MNS) research group, University of Amsterdam

²Parallel Computing Systems (PCS) research group, University of Amsterdam

³Complex Cyber-Infrastructure (CCI) research group, University of Amsterdam

TAs: Lex Bolt, Louis Boshoff, Tom Groot, Emo Maat, Tim Roepke, P.H.W. Wildschut

Dr. Zhiming Zhao (z.zhao@uva.nl)

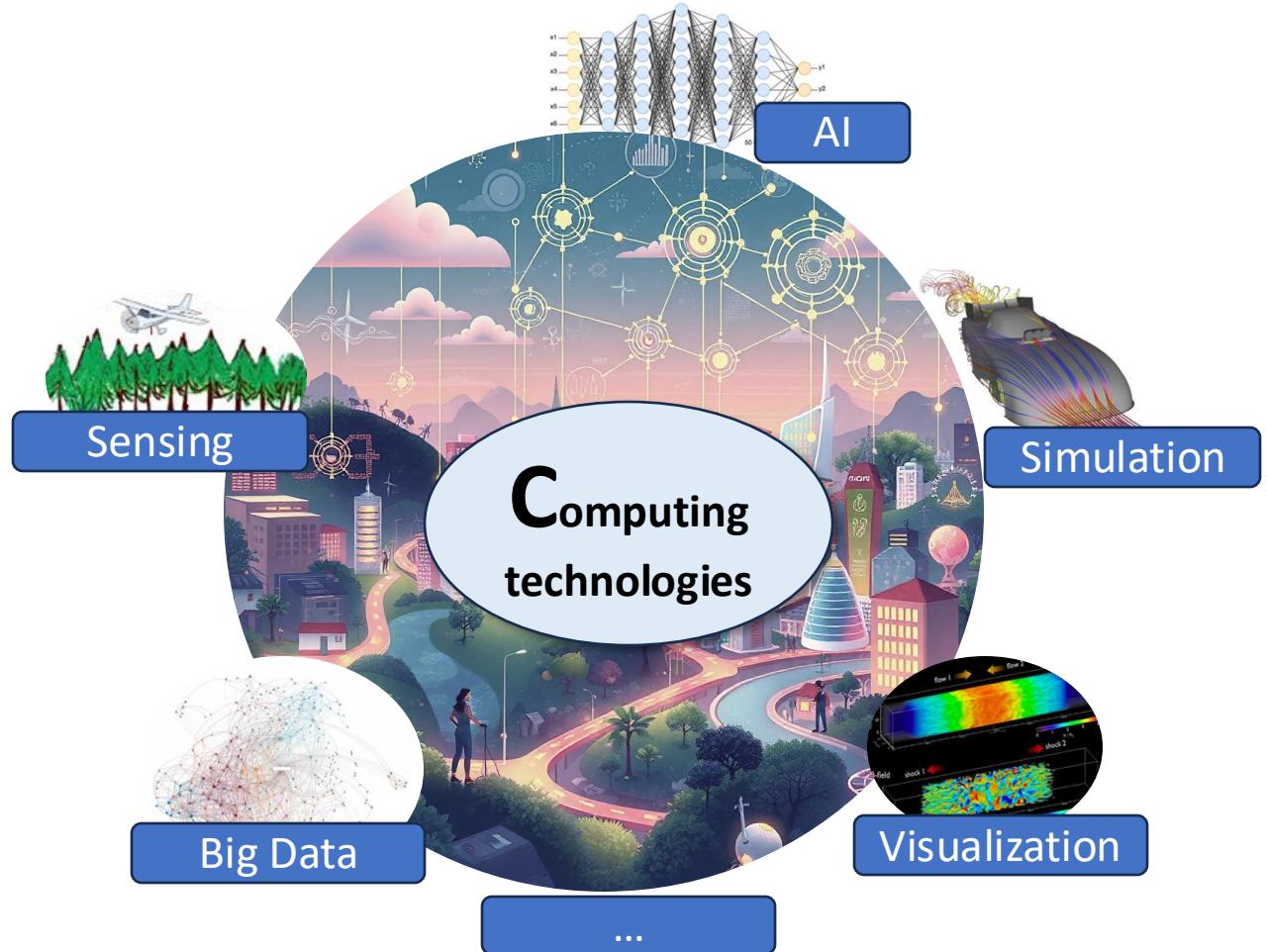
Associate Professor

MultiScale Networked Systems, UvA,

- Research areas
 - Distributed systems and cloud computing
 - Time-critical cloud application and infrastructure optimization
 - Big data management, scientific workflow, and data-intensive systems
 - Blockchain, Decentralized marketplaces
- Teaching and Education
 - Distributed and Parallel Programming (CS bachelor)
 - DevOps and Cloud-based Software Development (SE master)
 - Cloud and web services (CS master)
 - Graduation projects
- Contact
 - <https://staff.fnwi.uva.nl/z.zhao/>

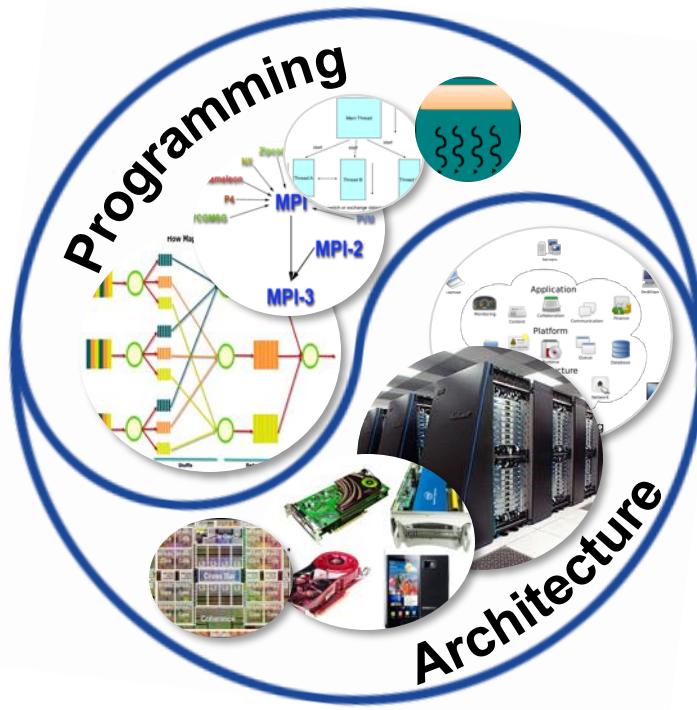
About the course

- Computing challenges in digital innovations
 - Artificial intelligence
 - Big data
 - Computing technologies
- Computing technologies
 - Speed
 - Scale
 - Reliability



1. Distributed and parallel programming

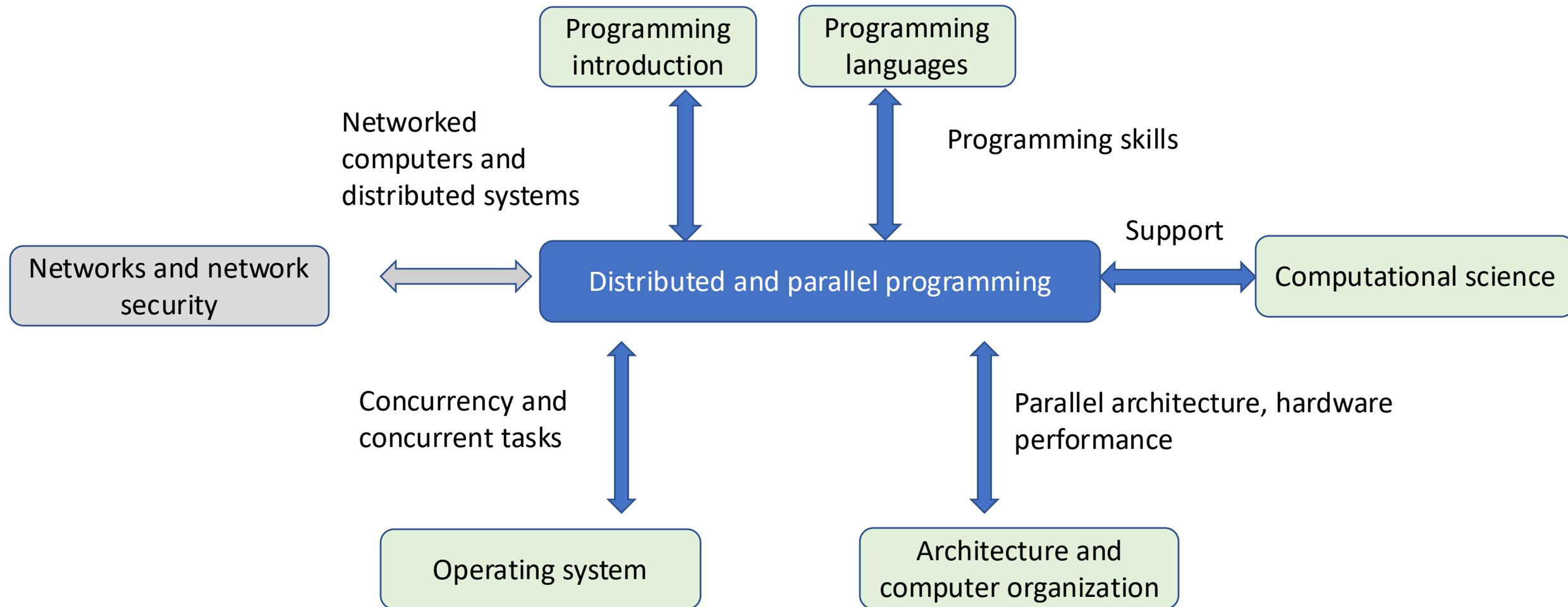
DPP is about the **architecture** and **programming models** of modern advanced **computing systems**.



Programming paradigm

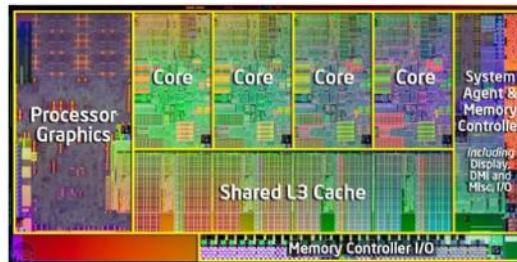
Performance analysis

Relation to the other courses



Changes in computing systems

- From single core to multi-core and many cores
- From shared memory to distributed memory
- From sequential execution to concurrent, parallel and distributed executions



Quad-core processors such as this Intel Sandy Bridge introduce programming complexities not present in a single-core CPU.



2. Structure of the content

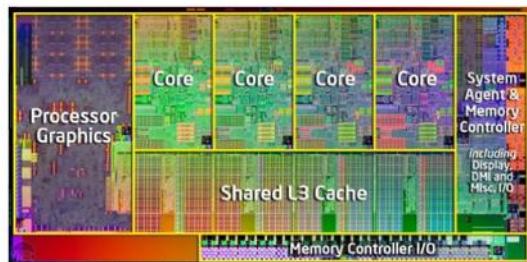
- **Chapter 1:** Parallel computing introduction
- **Chapter 2:** Multi-threading
- **Chapter 3:** GPU Programming
- **Chapter 4:** Message passing interface
- **Chapter 5:** Distributed programming paradigms and applications

Week1: Chapter 1

Parallel computing introduction



- **Topics:**
 - From parallel to concurrency and parallelism
- **Assignment:**
 - Lab assignment 0: DAS tutorial
 - Literature study



Quad-core processors such as this Intel Sandy Bridge introduce programming complexities not present in a single-core CPU.

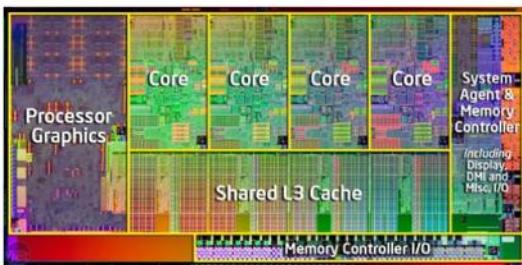


Week1-2: Chapter 2

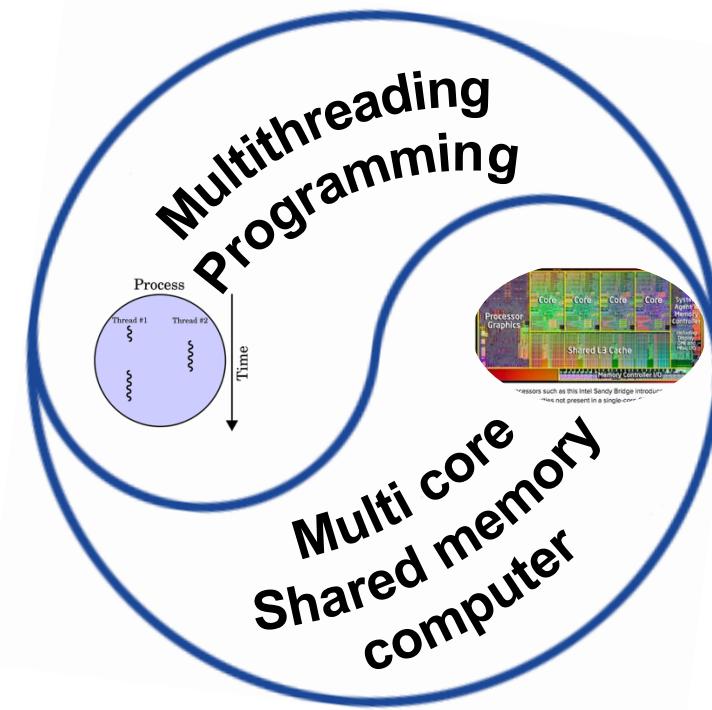
Multithread programming



- **Topics:**
 - Architecture: shared memory
 - Programming paradigm: multi-threading
- **Assignment:**
 - Lab assignment 1: multi-threading



Quad-core processors such as this Intel Sandy Bridge introduce programming complexities not present in a single-core CPU.

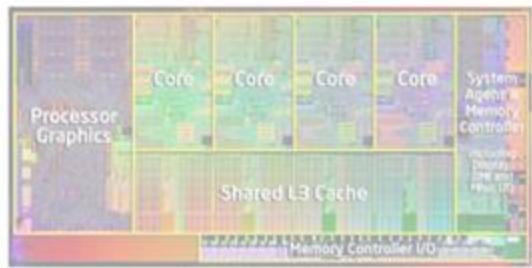




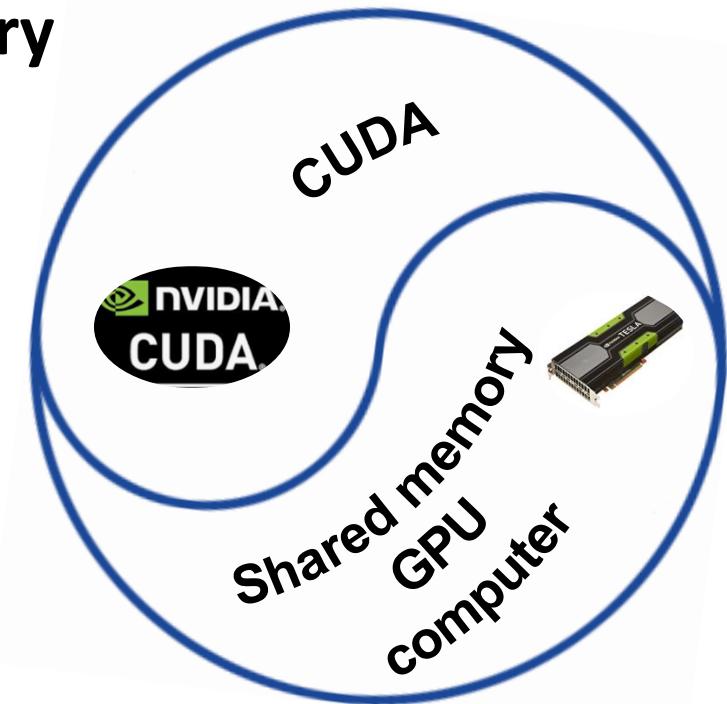
Week 2-4: Chapter 3

GPU programming

- **Topics:**
 - Architecture: GPU, Many-core shared memory
 - Programming paradigm : CUDA
- **Assignment:**
 - Lab assignment 2: CUDA



Quad-core processors such as this Intel Sandy Bridge introduce programming complexities not present in a single-core CPU.

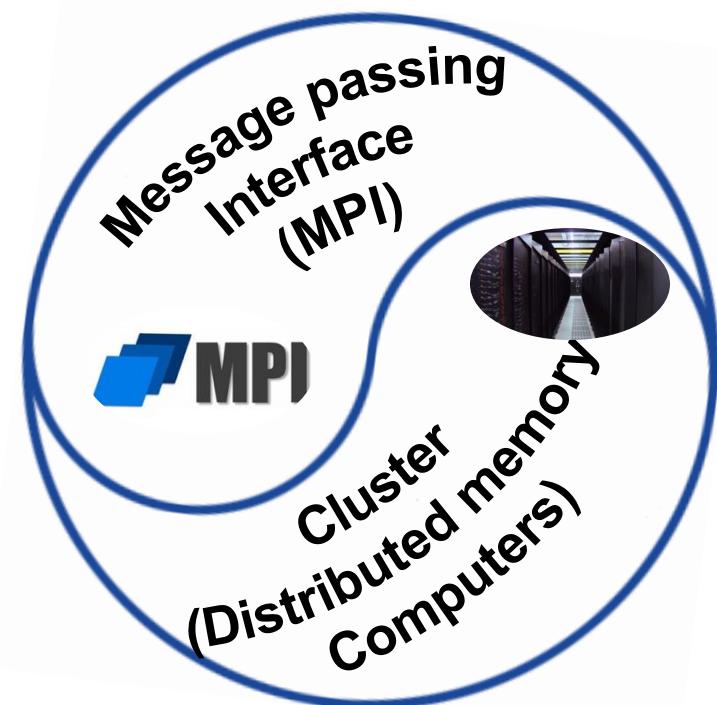
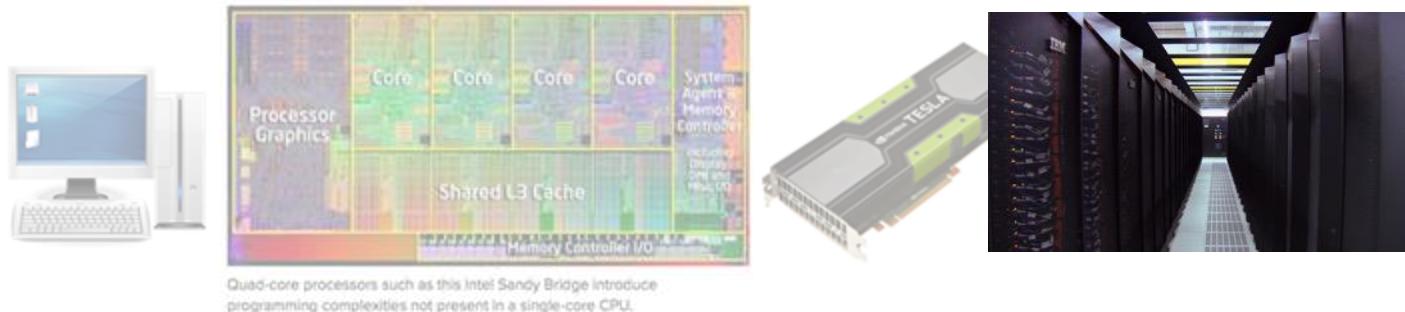


Week 5-6: Chapter 4

Message passing interface



- **Topics:**
 - Architecture: **Distributed memory (cluster)**
 - Programming paradigm: Message Passing Interface (MPI)
- **Assignment:**
 - Lab assignment 3: MPI



Week 6-7: Chapter 5

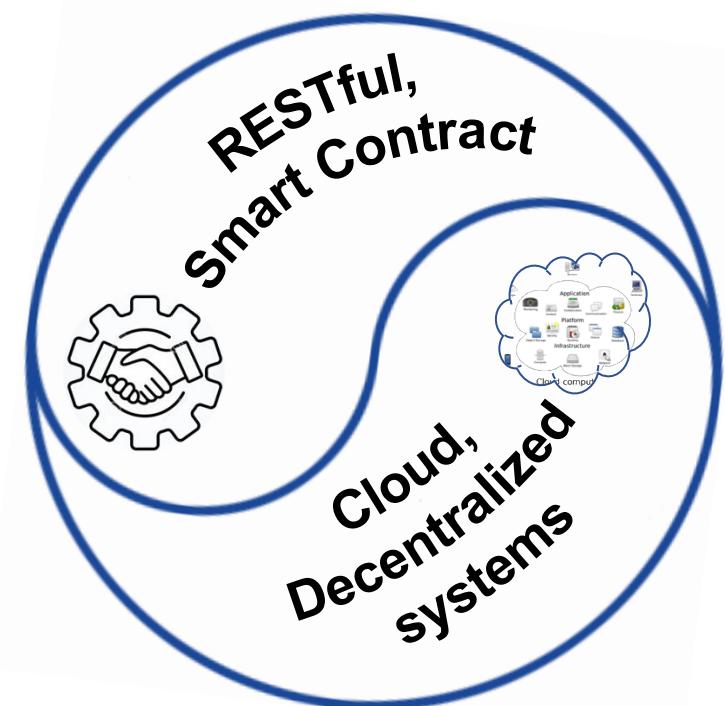
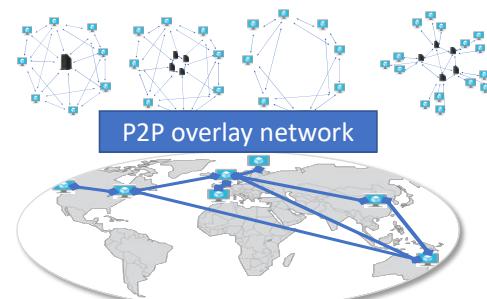
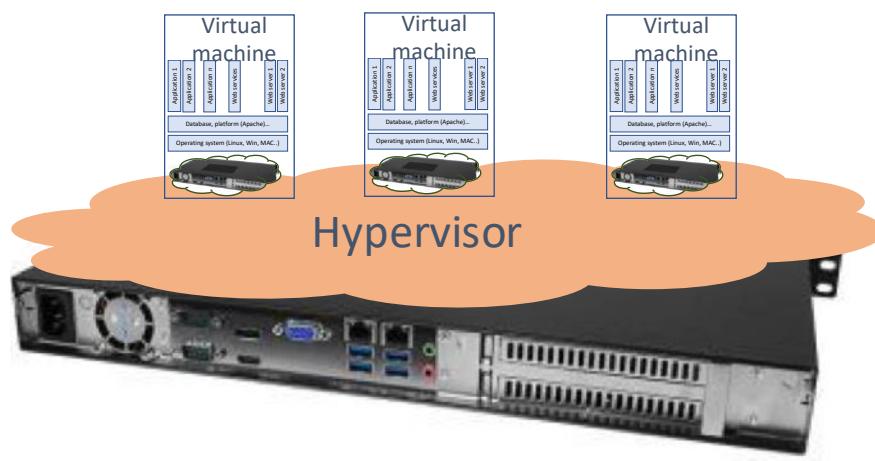
Cloud, big data, and DApp

- **Topics:**

- **Architecture:** Virtualization technologies and Cloud IaaS, P2P, and Blockchain
- **Programming paradigm:** RESTful services, containers and big data platforms, Smart contract

- **Assignments:**

- Lab assignment 4: Tutorial on Cloud, big data and blockchain



Structure of the content

- **Chapter 1:** Parallel computing introduction
- **Chapter 2:** Multi-threading
- **Chapter 3:** GPU Programming
- **Chapter 4:** Message passing interface
- **Chapter 5:** Distributed programming paradigms and applications

3. Objectives, methods and assessment

1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
2. **Explain and describe** basic programming models for parallel and distributed computing systems.
3. **Get familiar with** the programming models of cloud and decentralized systems.
4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.

Activities

1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
2. **Explain and describe** basic programming models for parallel and distributed computing systems.
3. **Get familiar with** the programming models of cloud and decentralized systems.
4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.

All Lectures

Activities

1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
2. **Explain and describe** basic programming models for parallel and distributed computing systems.
3. **Get familiar with** the programming models of cloud and decentralized systems.
4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.

5 Lab assignments in 7 weeks

Activities

1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
 2. **Explain and describe** basic programming models for parallel and distributed computing systems.
 3. **Get familiar with** the programming models of cloud and decentralized systems.
 4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
 5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
 6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.
- } Literature study
(academic skill)

Assessment

1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
2. **Explain and describe** basic programming models for parallel and distributed computing systems.
3. **Get familiar with** the programming models of cloud and decentralized systems.
4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.

Exam (25%)

- Remember/Understand

Lab (45%)

- apply/analysis

LS report (30%)

- apply/analysis

Interaction and Feed back

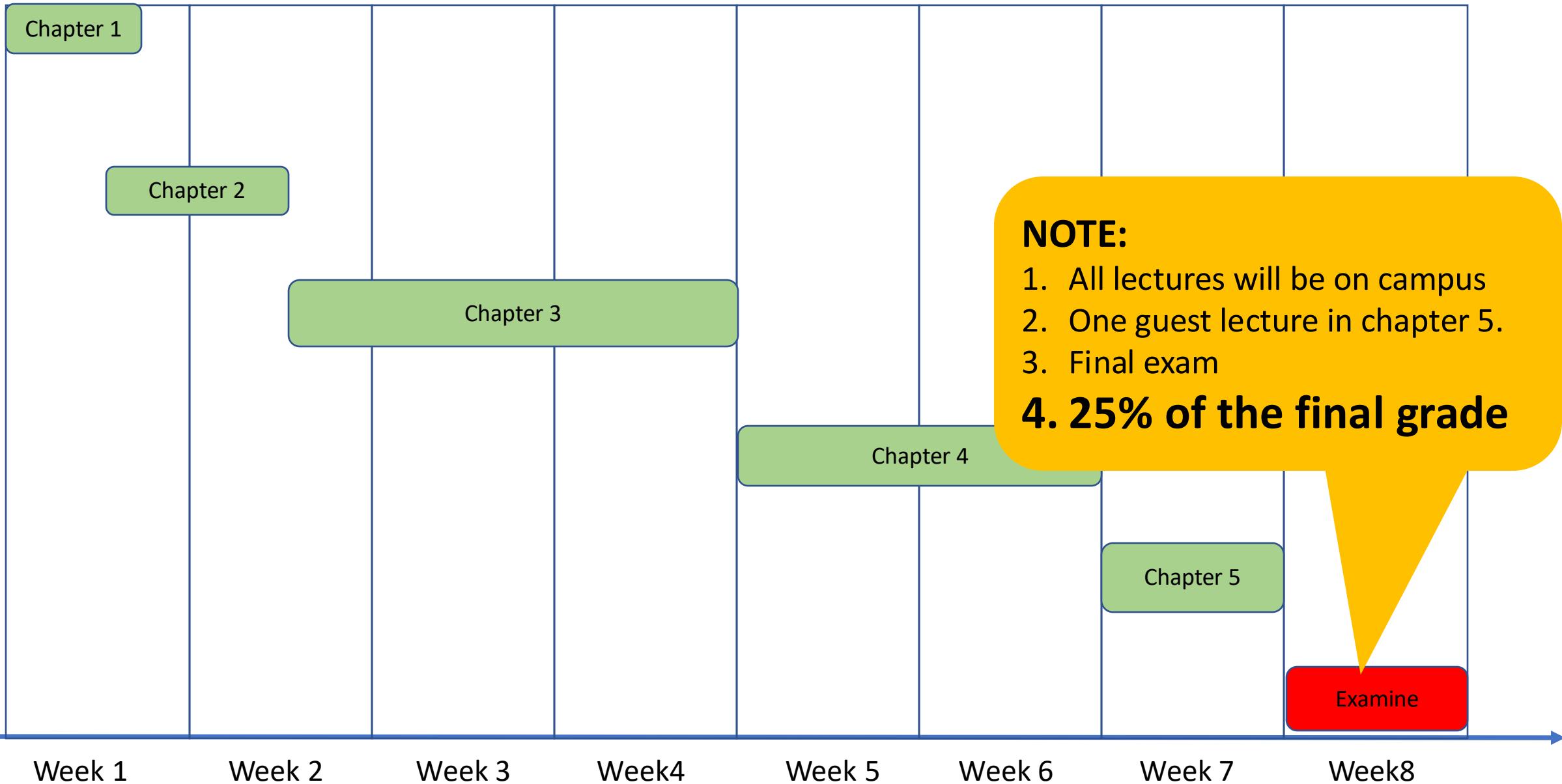
1. **Explain and describe** the taxonomy of parallel and distributed systems, and the role and functionality of the different components that appear in these computing systems.
2. **Explain and describe** basic programming models for parallel and distributed computing systems.
3. **Get familiar with** the programming models of cloud and decentralized systems.
4. **Program** small-scale applications using the main concepts of parallel processing and programming models of Pthreads, OpenMP, CUDA, and MPI.
5. **Analyze** the performance results of parallel and distributed systems and evaluate their results.
6. **Conduct** a literature study on a selected topic of parallel and distributed computing systems.

Classroom Quiz,
discussion,
Email, CANVAS

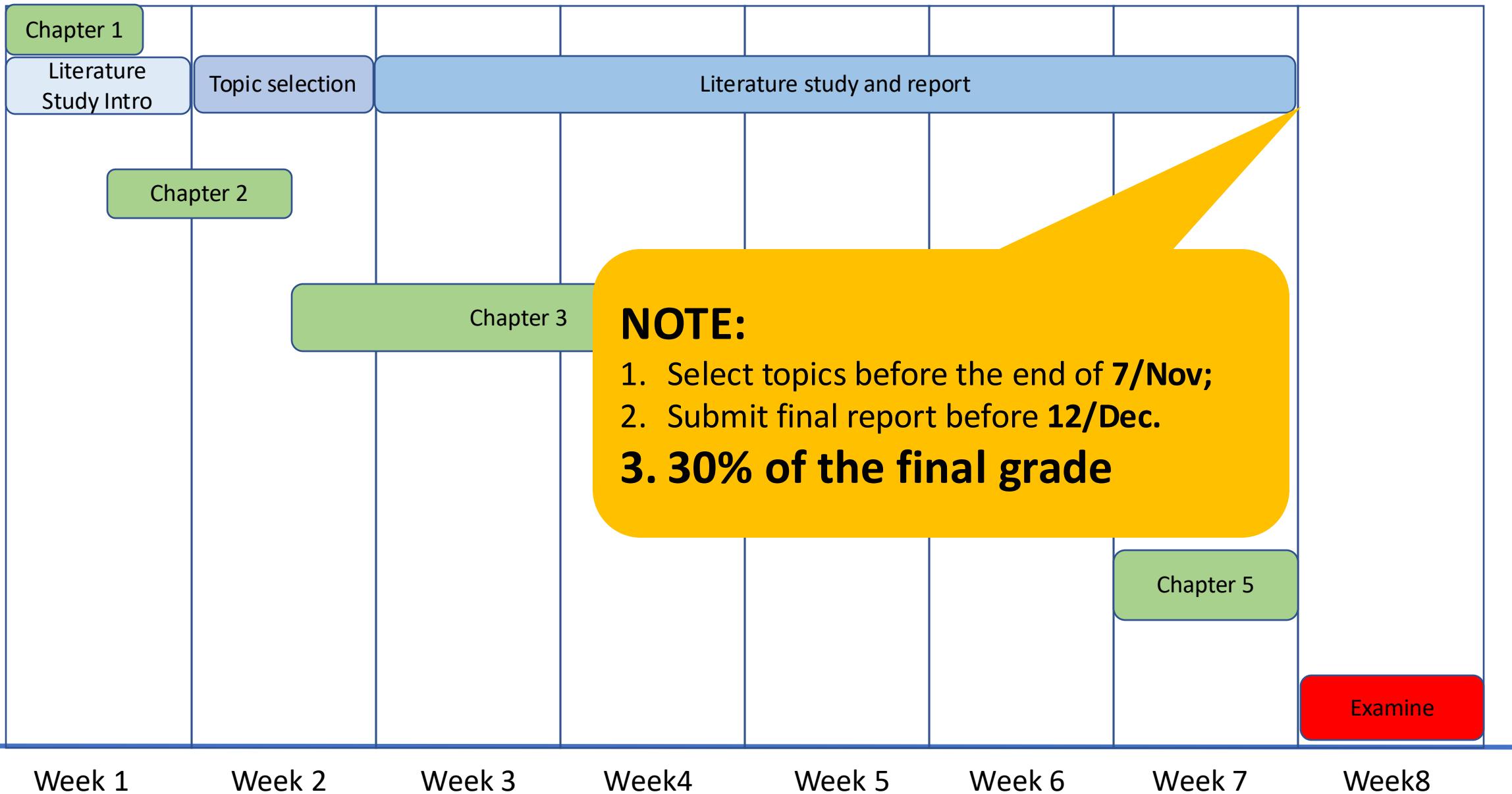
Dedicated TA
per group,
...

Teacher

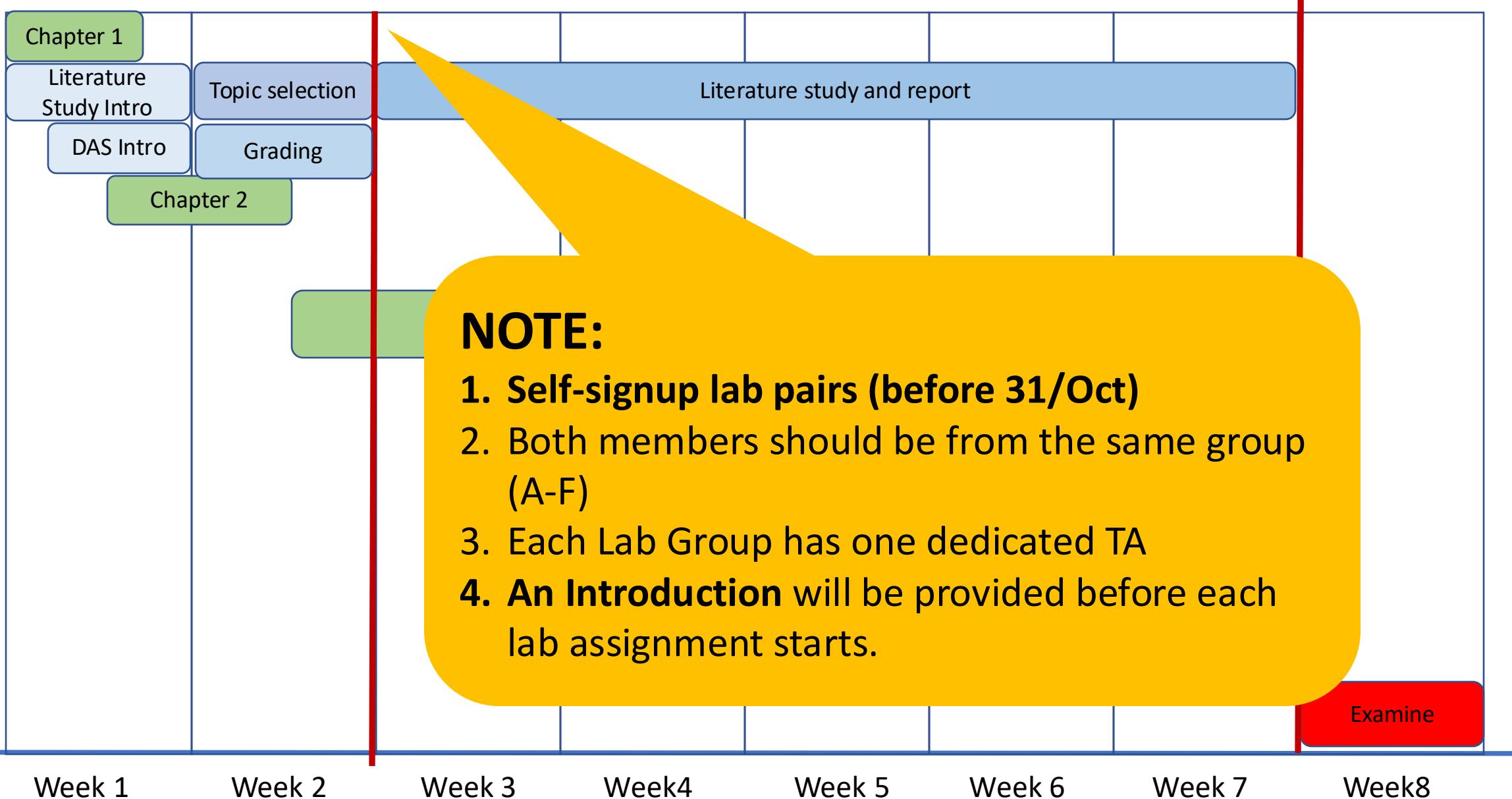
4. Schedule of DPP 2025



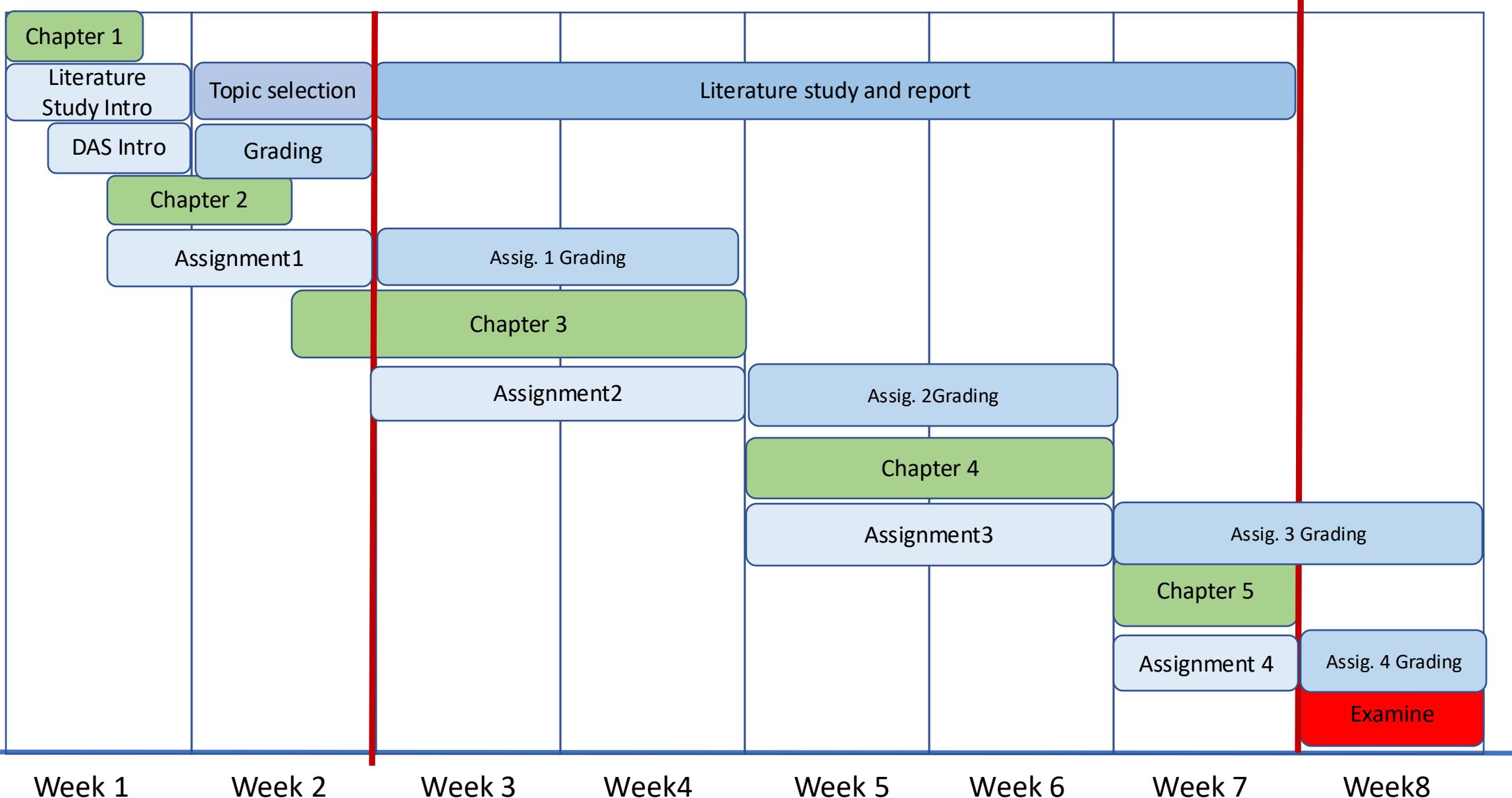
4. Schedule of DPP 2025



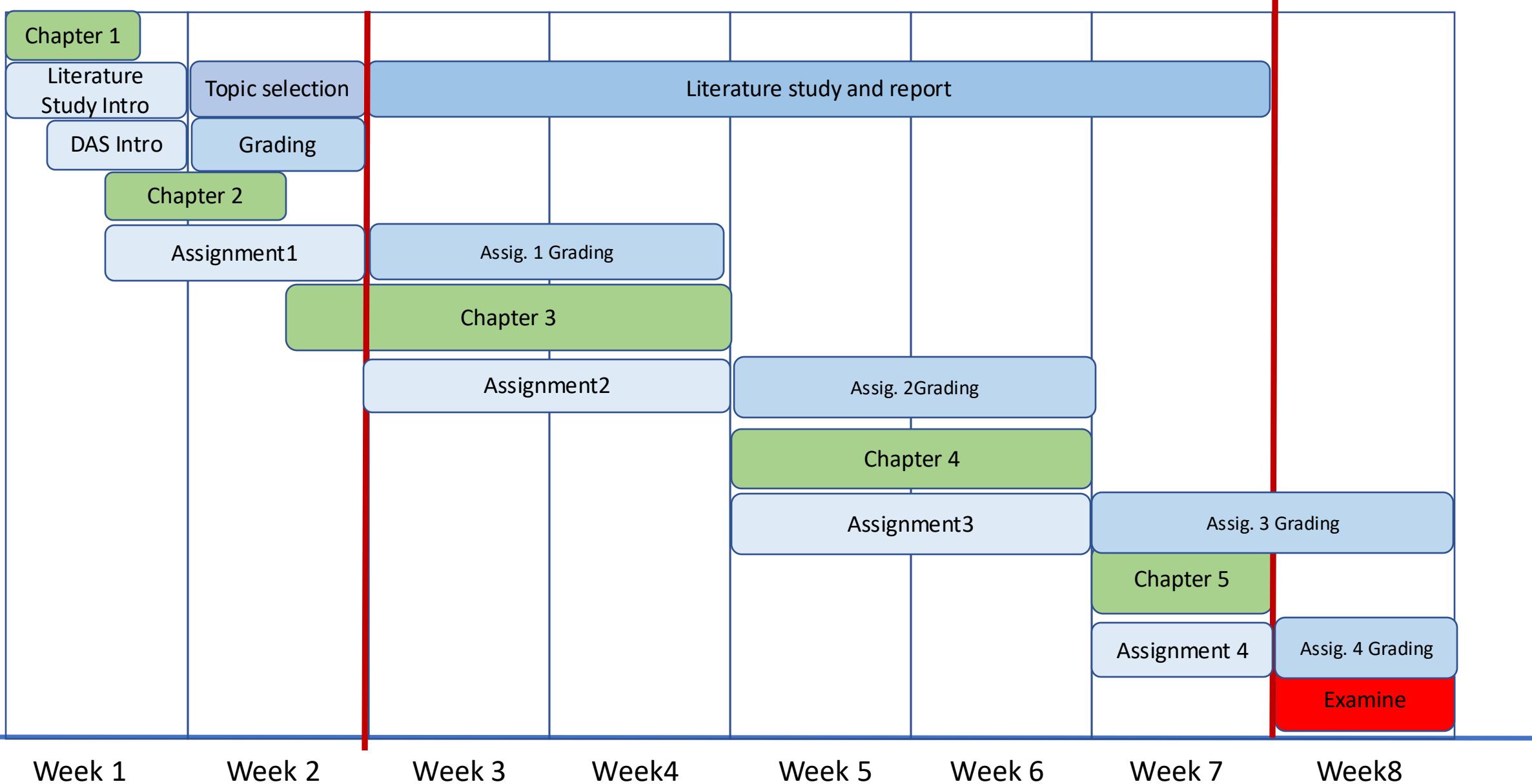
4. Schedule of DPP 2025



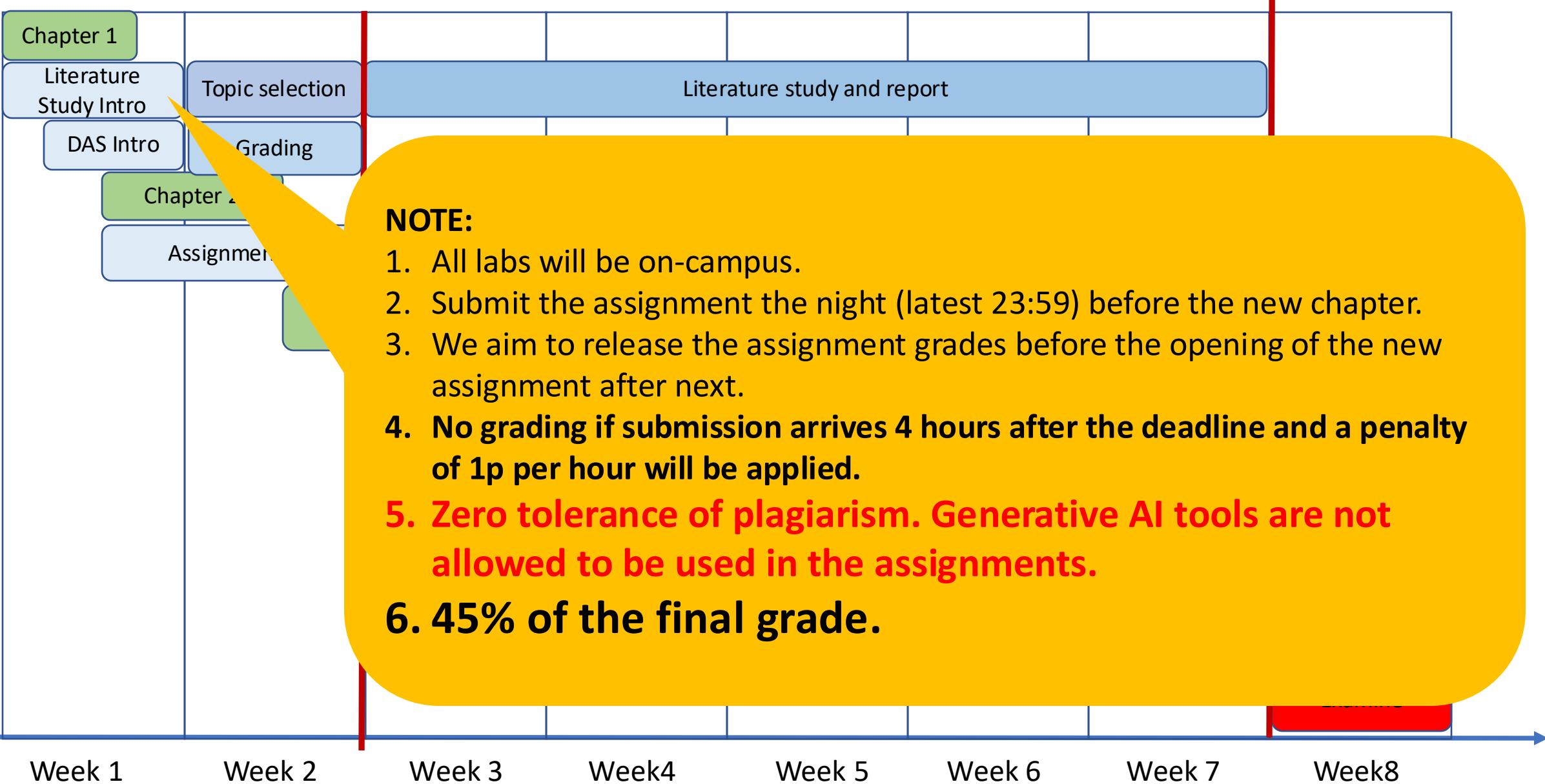
4. Schedule of DPP 2025



4. Schedule of DPP 2025



4. Schedule of DPP 2025



5. Summary

- Contact: dpp@list.uva.nl
 - Dr. Zhiming Zhao: Z.Zhao@uva.nl
 - Dr. Dolly Sapra D.Sapra@uva.nl
 - Dr. Adam Belloum: A.S.Z.Belloum@uva.nl
- Wednesday morning (29/October), 11:00-13:00, an introduction to literature study (C0.05)
- Please make your Lab Group ASAP; random team assignment by 31/Oct.

Home
Announcements
Assignments
Grades
Discussions
Modules
Course manual
People
Course Catalogue
Zoom
Quizzes
Rubrics
Media Gallery
Pages
Files
Syllabus
Outcomes
Conferences
Collaborations
Settings

Literature study topics

The topics to select from are listed in below:

1	Energy benchmarking for applications (case-studies)
2	Energy-efficiency for parallel and/or distributed systems (case-studies and/or techniques)
3	Energy efficiency metrics and benchmarking tools for parallel systems/applications
4	Dark silicon in multicore
5	Heterogeneous vs Homogeneous systems (multi-cores, multi-processors, multi-nodes)
6	Domain-specific and/or application-specific parallel/distributed/heterogeneous architectures/systems
7	Challenges in exascale computing (architectures)
8	The role of Co-design in exascale applications
9	Amdahl's Law for modern systems
10	Performance metrics in parallel computing
11	Performance metrics in parallel and/or distributed architectures
12	Automated parallelization: problems and solutions ("from sequential to parallel without user intervention")
13	The memory wall: problems and solutions
14	The power wall: problems and solutions
15	The ILP wall: problems and solutions
16	The use and/or the end of Moore's Law
17	The impact of SIMD on performance (for different architectures)
18	Single-chip heterogeneous systems: case-studies for different application (domains)
19	GPUs in embedded systems
20	A comparative analysis of applications running on a Top500 machine
21	A comparative analysis of 2 (different) architectures from top 10 in Top500 and Green500
22	Convergence of HPC and BigData (HPDA) and AI architectures/infrastructure

5062COPP6Y > People > Groups

2020/21 Sem. 1, per. 2	
Everyone	Groups
<input type="button" value="Search groups or people"/>	
▶ GroupA-1 Group-Assignment	2 students
▶ GroupA-2 Group-Assignment	2 students
GroupA-3 Group-Assignment	0 students
GroupA-4 Group-Assignment	0 students
GroupA-5 Group-Assignment	0 students