Project Assigment

Project details

The purpose of the project is to apply data parallelism and CUDA concepts

- Optimize performance in the deployment of algorithms / applications on GPUs
- Development of optimized libraries for GPUs
- Effective development of applications / systems using GPUs

Project details

The purpose of the project is to apply data parallelism and CUDA concepts

- Optimize performance in the deployment of algorithms / applications on GPUs
- Development of optimized libraries for GPUs
- Effective development of applications / systems using GPUs

Active engagement with a poblem to build knowledge or develop skills

Develop communication skills (writing and oral)

Develop team work skills

Project details

The purpose of the project is to apply data parallelism and CUDA concepts

- Optimize performance in the deployment of algorithms / applications on GPUs
- Development of optimized libraries for GPUs
- Effective development of applications / systems using GPUs

Examples

- Thorough performance analysis and improvement of GPU code or architectures
- Implement a GPU version of some existing computationally intensive CPU code
- Reproduce some existing GPU research work
- Do novel GPU research

Project details

Relevant rules:

- Project proposal Pitch
- Proposal
- Report
- Presentation

Project details

Relevant rules:

- **Project proposal Pitch**
  - **Date:** November 26-27, 2025
  - **Time:** 3 mins
  - 2 slides with the main idea and targets of the project
  - Fill the online form to select any of the available slots (individual, groups of 2/3 students)
- Proposal
- Report
- Presentation

3 only if the project is particularly complex

Project details

Relevant rules:

- **Project proposal Pitch**
  - **Date:** November 26-27, 2025
  - **Time:** 3 mins
  - 2 slides with the main idea and targets of the project
  - Fill the online form to select any of the available slots (individual, groups of 2/3 students)
- Proposal
- Report
- Presentation

3 only if the project is particularly complex

- Wait for a go/no go action from the professors
- We will check if multiple groups have done the same proposal, or if the proposal is sound
- Maximum after 2 weeks
- In case of no go, propose a new topic and submit the proposal

Project details

Relevant rules:

- Project proposal Pitch
- **Proposal**
  - **Date:** Submitted at least 6 weeks before the presentation day
  - Winter session exams: **01/02/2026 – 22/02/2026**
  - Summer session exams: **08/06/2026 – 18/07/2026**
  - 1 page with an abstract of the project:
    must include:
    - Title of the project
    - Authors
    - Main problem
    - Targets or objectives of the assignment (KPIs, milestones, workflow)
    - References
- Report
- Presentation

Project details

Relevant rules:

- Project proposal Pitch
- **Proposal**
  - **Date:** Submitted at least 6 weeks before the presentation day
  - Winter session exams: **01/02/2026 – 22/02/2026**
  - Summer session exams: **08/06/2026 – 18/07/2026**
  - 1 page with an abstract of the project:
    must include:
    - Title of the project
    - Authors
    - Main problem
    - Targets or objectives of the assignment
    - References
- Report
- Presentation

Deadline	Exam dates (4 sessions)
December 24th / 2025	e.g., February 4th
January 6st	e.g., February 17th
April – May	June – July
July	September 1st – 20th

Project details

Relevant rules:

- Project proposal Pitch
- **Proposal**
  - **Date:** Submitted at least 6 weeks before the presentation day
  - Winter session exams: **01/02/2026 – 22/02/2026**
  - Summer session exams: **08/06/2026 – 18/07/2026**
  - 1 page with an abstract of the project: must include:
    - Title of the project
    - Authors
    - Main problem
    - Targets or objectives of the assignment
    - References
- Report
- Presentation

Deadline	Exam dates (4 sessions)
December 24th / 2025	e.g., February 4th
January 6st	e.g., February 17th
April – May	June – July
July	September 1st – 20th

- If you miss the winter deadlines, you cannot apply for project presentation in those sessions
- The same happens in case of a change of the project after the winter deadlines

Project details

Relevant rules:

- Project proposal Pitch
- Proposal
- **Report**
  - **Date:** submitted at **least one week before the presentation day** (e.g., winter, summer exam session)
  - 10-12 pages, IEEE 2-column, times new roman, 10pf
    https://www.ieee.org/conferences/publishing/templates
    Must include: (don't be verbose or superficial)
    - Introduction, brief background, methodology / proposed method
    - Experimental results and analysis of results (meaningful figures and pictures)
    - Discussion of the results
    - References
  - Technically sound, clear, organized, and well-written
- Presentation

Project details

Relevant rules:

- Project proposal Pitch
- Proposal
- Report
- **Presentation**
  - **Date:** Slides and project codes submitted at least one week before the presentation day
  - **Time:** 15 mins + 5 mins Q/A
  - don’t be verbose or superficial
  - **Functionality and performance:**
    - Produces correct results
    - Achieves good speedup relative to base code and/or competitors
    - Optimizations applied
  - **Code Quality: (GitHub link is strongly advised)**
    - Clear coding
    - Well documented

Project details

Intermediate check:

Starting from December 17, we will organize roundtable activities

- Focused on specific group activities
- Everyone can join the discussion, share ideas, propose solutions…

We will identify fixed slots with the groups

- We will interact with you and allocate them according to everyone's availability

If you want to apply for the winter exam sessions, you must attend at least one of the intermediate checks

Project details

Example of application optimization:

Step	Brief outline	Concrete example
1	Choose an application	Dense Matrix-Matrix Multiply
2	Determine what part of the application is taking the majority ofthe time	Algorithm Evaluation through Analysis/Simulation
3	Determine one or more data-parallel approaches to solving the problem	Assign one output to each thread in a gather-style approach
4	Create multiple implementations of the approach	One naïve version, one version with shared memory tiling, one version with register tiling
5	Measure the performance and execution characteristics of the implementations for various parameters	Record memory transfer time, kernel time, utilization, FLOPS, etc
6	Relate results to course concepts	Performance may be impacted by utilization, shared-memory accesses, memory coalescing, and control divergence.

Project details

Example of project topics:

Physical simulations:
- e.g., Particles or bodies interaction simulations (N-body simulations)

Criptography:
- e.g., hardware-based cryptography (MEAA, SHA-3)

Communications:
- e.g., GPU-accelerated software routers

CAD tools:
- e.g., GPU-accelerated digital circuit logic simulators
- e.g., GPU-accelerated analog circuit simulators
- e.g., Provide GPU support to an existing CAD tool (language)

Artificial intelligence and machine learning:
- e.g., Low-level optimization of neural networks
- e.g., contribution or development of libraries for tensor operators

Project details

Example of project topics:

Analysis and research:

- e.g., Vibe coding for GPUs
- e.g., Analysis of software-based algorithms for sparse matrix multiplication operations in GPUs
- e.g., Custom CUDA Kernel Autotuner (automatic framework to explore kernel launch parameters block size, tiling, etc., for optimal performance)
- e.g., Hybrid CPU-GPU Sparse Tensor Library (Development of small library for sparse tensor operations optimized for structured sparsity)
- e.g., Parallel Conjugate Gradient (CG) Solver in CUDA

Fault tolerance and reliability:

- e.g., Analysis of parallel workloads to software errors, error correction, redundancy, memory coherency

Complex image processing and computer vision:

- e.g, GPU-accelerated multi-spectrum / multi-channel image / video processing algorithms

Project details

Example of project topics:

Embedded system development:

- e.g., GPU-powered integration of complex functionalities (time-constraints)

Any other domain that can take advantage of the GPU computational power

Other meaningful analysis on the use of GPUs

- Kyle Berney and Nodari Sitchinava. Eliminating Bank Conflicts in GPU Mergesort. In Proceedings of the 37th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA ‘25), 2025

Project details

Other examples of projects:

1. Heat flow in two-dimensional area
2. GPU parallel fault simulator Analog/Digital
3. Real-time Sterescopic vision (using videos)
4. Numpy/C++ extention of non-native computational formats (Cfloat8, Cfloat4, posit8, posit16, posit32) for GPUs
5. Systolic array cores simulation using a GPU
6. Thread redundancies (DWC, TMR), tested using benchmarks (e.g., Nvidia examples SDK, rodiania)
7. ABFT for Matrix multiplication for any size (library)
8. GPU stress benchmarking (verification using the profiler)
9. GPU-based neuromorphic circuit simulation