

# Project Contract

## Project Title

**Understanding Lasso Through Offline Memory Checking, Spartan, and Spark**

## Overall Goal

The primary goal of this project is to develop a clear and comprehensive understanding of **Lasso**, a modern SNARK construction used that enables efficient lookup arguments, notably used in **Jolt**. To achieve this, the project will trace the conceptual lineage of Lasso by studying the systems it builds upon:

- **Offline Memory Checking**, which underlies parts of the verifiable computation model used in these systems.
- **Spark**, introduced as a component of Spartan,
- **Spartan**, which provides foundational ideas and IOP structure,

The project culminates in a written report synthesizing how these components fit together and how they ultimately inform the design of Lasso. Work from another course will explain further priors like Sumcheck, GKR and Multilinear Extensions. Although depending on what's agreed with the other course's supervisor some of these topics may be included in this course. The dividing line should of course be made perfectly clear to this course's supervisor and examiners.

## Scope

### 1. Explain the Foundations: Offline Memory Checking

- Define the offline memory-checking model and its relevance to verifiable computation.
- Explain how memory-checking allows verification of RAM programs with minimal interaction.
- Summarize known constructions and complexity results.
- Highlight how these ideas motivate aspects of Spartan and Spark.

### 2. Study and Summarize Spartan

- Focus only on the parts of Spartan relevant to Lasso.
- Explain at a high-level what Spartan is and how it utilizes Spark to create a fast SNARK.

### 3. Explain Spark (from the Spartan paper)

- Summarize Spark's role in Spartan.
- Explain how Spark uses a simple SNARK construction, based on GKR, and offline memory-checking
- Describe how Spark introduces constructs that Lasso later generalizes or adapts.

### 4. Understand and Explain Lasso

- Provide a complete conceptual explanation of the Lasso proof system.
- Describe its polynomial IOP structure and why it enables efficient lookup arguments.
- Explain how Lasso builds on Sparkn.
- Explain at a high-level how this is used in Jolt.

### 5. Produce a Final Report

- The final deliverable is a comprehensive, well-organized report that includes:
  - Formal system summaries,
  - Diagrams and tables,
  - Conceptual comparisons,
  - A clear explanation of how Lasso emerges from Spark and Spartan,
  - Connections to offline memory-checking,
  - Discussion of strengths, limitations, and open questions.

## Expected Outcomes

- A detailed written explanation of Offline Memory Checking, Spark, and Lasso.
- A clear articulation of how these systems relate and evolve into one another.
- A strong conceptual foundation for understanding Lasso and related SNARKs.

- Ideally, the final document should function as a go-to reference for catching up on how Lasso, and therefore Jolt, work. This document should be easier to read than the original papers, for which the primary audience is existing researchers.

## Methodology

- **Research & Reading**

- Study the primary papers:
  - Offline memory checking literature,
  - Spartan,
  - Spark,
  - Lasso.

- **Synthesis & Explanation**

- Extract the core ideas and abstractions.
- Organize findings into a coherent narrative.

- **Evaluation Criteria**

- Clarity and correctness of explanations,
- Depth of understanding in the final report.

## Conclusion

This project aims to provide a deeper conceptual understanding of **Lasso** by tracing and explaining its lineage. By studying **offline memory checking**, **Spartan**, and **Spark**, the report will clarify the motivations, mechanisms, and innovations that culminate in Lasso. The final deliverable will be a structured and insightful reference on the foundations and evolution of modern SNARK design.