

**Algorithmic Algebraic Parser
Software Development Plan
Version 1.9**

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

Revision History

Date	Version	Description	Author
<dd/mmm/yy>	<x.x>	<details>	<name>
2/2/26	1.1	Added content to sections 1.3, 1.4, 2.1, 2.2, 2.3	Simeon Geib
2/3/26	1.2	Added content to sections 4.2.2, 4.2.3, 4.2.4	Nathan Philips
2/5/26	1.3	Added content to sections 3.1, 3.2	Spencer Weishaar
2/9/26	1.4	Updated content in sections 3.1, 3.2	Spencer Weishaar
2/10/26	1.5	Document formatting changes, remove 3.1 for redundancy	Shlok Prabhu
2/10/26	1.6	Updated content in section 1.3. Added UPEDU as an acronym.	Spencer Weishaar
2/16/26	1.7	Added content to sections 1.1, 1.2, 1.5, 3.1	Zema Samuel
2/17/26	1.8	Updated content in 3.1	Lincoln Earnshaw
2/17/26	1.9	Added content to project schedule, 4.1.3	Spencer Weishaar

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

Table of Contents

1. Introduction	4
1.1 Purpose	4
1.2 Scope	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 References	4
1.5 Overview	4
2. Project Overview	6
2.1 Project Purpose, Scope, and Objectives	6
2.2 Assumptions and Constraints	6
2.3 Project Deliverables.....	6
2.4 Evolution of the Software Development Plan	6
3. Project Organization	8
3.1 Roles and Responsibilities - Spencer.....	9
4. Management Process	11
4.1 Project Plan.....	11
4.1.1 Iteration Objectives	11
4.1.2 Releases.....	11
4.1.3 Project Schedule.....	11
4.2 Project Monitoring and Control.....	11
4.2.1 Quality Control.....	11
4.2.2 Risk Management.....	11
4.2.3 Configuration Management.....	12
5. Annexes.....	13

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

Software Development Plan

1. Introduction

1.1 Purpose

The purpose of this Software Development Plan is to define the process, schedule, responsibilities, and management procedures used to organize and control the development effort for the Arithmetic Expression Evaluator project. The SDP will be used to track project progress and coordinate team activities throughout the project lifecycle.

The project manager will use this document to plan work, allocate tasks, and monitor milestone completion. Team members will use it to understand assigned responsibilities, task dependencies, and deadlines.

The plan also establishes procedures for requirements management, configuration control, quality assurance, and risk management to ensure the final project deliverables satisfy the course requirements.

1.2 Scope

This Software Development Plan applies to the Arithmetic Expression Evaluator project and governs all project activities including planning, implementation, testing, and documentation. It defines how the project will be organized, scheduled, and monitored during development.

This document establishes team responsibilities, identifies project artifacts, and provides methods for tracking progress throughout the project lifecycle.

1.3 Definitions, Acronyms, and Abbreviations

There currently exist no definitions, acronyms, or abbreviations. In the event any are added, a “Project Glossary” document will be created.

- UPEDU - The Unified Process for EDUcation is a customized, web-enabled version of the set of best practices for software engineering, using RUP (Rational Unified Process). It provides an iterative, phase-based framework to guide students through development cycles.
- SDP-PROC - Software Development Process
- SDP-PLAN - Software Development Plan

1.4 References

There currently exist no additional references outside of this document. As they are created, they will be referred to here.

1.5 Overview

This Software Development Plan contains the following information:

Project Overview	—	Provides a description of the project’s purpose, scope, and objectives.
------------------	---	---

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

- Project Organization — Describes the organizational structure of the project team.
- Management Process — Explains the estimated schedule, defines major phases and milestones, and describes how the project will be monitored.
- Applicable Plans and Guidelines — Provides an overview of the software development process, including methods, tools, and techniques to be followed.

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

This project aims to create an arithmetic expression parser component that will be used in a compiler for the upcoming language “L”. The primary objective is to systematically develop this parser and deliver it, along with its related documentation, by May 7th, 2026.

The expected deliverables are as follows:

- ❖ The Project Plan document
- ❖ The Software Requirements Specification document
- ❖ The Software Architecture document
- ❖ The Implementation of the parser
- ❖ The Test Cases Document
- ❖ The User Manual

2.2 Assumptions and Constraints

Our primary assumptions are as follows:

- ❖ The members stated in section 3.3 are the only members of this project.
- ❖ The members will not spend any of their own money on this project.
- ❖ The final implementation and remaining deliverables will be due by May 7th, 2026.
- ❖ There will be no changes to the already given project specifications.
- ❖ The program should be in C++.

2.3 Project Deliverables

There will be six deliverables in total:

Name	Delivery Date	Type
Project Plan	Feb 22	Document
Software Requirements Specification	Mar 15	Document
Software Architecture	Apr 5	Document
Implementation	May 7	Code
Test Cases	May 7	Document
User Manual	May 7	Document

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

2.4 Evolution of the Software Development Plan

The *Software Development Plan* will be revised prior to the start of each Iteration phase.

See **Revision History** for details.

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

3. Project Organization

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

3.1 Roles and Responsibilities - Spencer

Person	Unified Process for EDUcation Role
Lincoln Earnshaw	Implementors, Tester, Reviewer
<ul style="list-style-type: none"> ❖ Email: 1008e025@ku.edu ❖ Github: https://github.com/LincolnEarnshaw ❖ Computing Platform Experience: VS, Scratch ❖ Programming Language Knowledge: Python, C, C++, Lua, HTML, CSS, JavaScript 	
Simeon Geib	Integrator, Tester, Implementors
<ul style="list-style-type: none"> ❖ Email: simeongeib@ku.edu ❖ Github: Emrillion ❖ Computing Platform Experience: Scratch ❖ Programming Language Knowledge: Python, Lua, Luau, C++ 	
Nathan Phillips	Integrator, Designer, Implementors, Change Control Manager
<ul style="list-style-type: none"> ❖ Email: nathan.s.phillips@ku.edu ❖ Github: https://github.com/NSP13737/ ❖ Computing Platform Experience: Apple IIe, Scratch, Code.org, ❖ Programming Language Knowledge: Apple BASIC, C++, Python, Unity 	
Shlok Prabhu	Project Manager, Analyst, Implementers
<ul style="list-style-type: none"> ❖ Email: shlokprabhu@ku.edu ❖ Github: shlockie-np ❖ Computing Platform Experience: Windows, VS, IntelliJ, Node.js, React ❖ Programming Language Knowledge: Python, Java, C++, C, HTML/CSS/JS/TS 	
Spencer Weishaar	Analyst, Tester, Implementors
<ul style="list-style-type: none"> ❖ Email: spencer.weishaar@ku.edu ❖ Github: sbw22 ❖ Computing Platform Experience: 	

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

❖ Programming Language Knowledge: Python, HTML/CSS/JavaScript, C	
Zema Samuel	
<ul style="list-style-type: none"> ❖ Email: zemasamuel@ku.edu ❖ Github: zemasamuel ❖ Computing Platform Experience: Windows, VS ❖ Programming Language Knowledge: Python, C++, HTML/CSS, JS 	Designer, Implementors

Team Availability:

For team availability, please refer to When2Meet (<https://www.when2meet.com/?34665480-EvOM9>)

Anyone on the project can perform [Any Role](#) activities.

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

4. Management Process

4.1 Project Plan

4.1.1 Iteration Objectives

- ❖ Project Requirements
 - Create specific goals that the software must meet to be deemed successful
- ❖ Project Structure (Design)
 - Translate requirements into coding ideas with a planned implementation
 - Create file structure for easy and clear maintainability
- ❖ Coding
 - Technical implementation with functional code to meet requirements
- ❖ Testing
 - Create unit tests to formally verify all parts of code are functioning as intended

4.1.2 Releases

- ❖ Planned Release V0.01 [*beta*] | 02/21/26
 - MVP implementation of file structure for project

4.1.3 Project Schedule

Deliverable 1 – Project Plan – Feb 22nd
 Deliverable 2 – Software Requirements – Mar 15th
 Deliverable 3 – Software Architecture – Apr 5th
 Deliverable 4 – Implementation – May 7th
 Deliverable 5 – Test Cases – May 7th
 Deliverable 6 – User manual – May 7th
 Term Project Peer Evaluation – May 7th

4.2 Project Monitoring and Control

4.2.1 Quality Control

Defects will be recorded and tracked as Change Requests, and defect metrics will be gathered (see Reporting and Measurement below).

All deliverables are required to go through the appropriate review process, as described in the Development Case. The review is required to ensure that each deliverable is of acceptable quality, using guidelines and checklists.

Any defects found during review which are not corrected prior to releasing for integration must be captured as Change Requests so that they are not forgotten.

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

4.2.2 *Risk Management*

Risks will be identified in the Inception Phase using the steps identified in the RUP for Small Projects activity “Identify and Assess Risks”. Project risk is evaluated at least once per iteration and documented in this table.

4.2.3 *Configuration Management*

Appropriate tools will be selected that provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.

The Change Requests are reviewed and approved by one member of the project, the Change Control Manager role.

Algorithmic Algebraic Parser	Version: 1.9
Software Development Plan	Date: Feb 17, 2026
SDP-PLAN-2026-02-17	

5. Annexes

The project will follow the UPEDU process.

Programming Guidelines	\doc\SPD-GUIDE-xxxx-xx-xx.pdf
Setup Guide	\doc\SDP-SETUP-xxxx-xx-xx.pdf
Software Development Iteration Plan	\doc\SDP-PLAN-xxxx-xx-xx.pdf
Pull Request Template	\github\PULL_REQUEST_TEMPLATE.md
Bug Report Template	\github\ISSUE_TEMPLATES.bug_report.md
Feature Request Template	\github\ISSUE_TEMPLATES.feature_request.md