

Reference Stories No.	Story Pool	Story Source	Summary (25-50 Words)
1	1	EECS 581 - Project 1	Use PyGame to implement a status indicator ("Playing", "Win", "Loss") onto the user interface. Created a textbox that displayed "Status: {status}", where the status variable would be updated dynamically as moves are made.
2	1	EECS 581 - Project 1	Implement an algorithm for minesweeper to place a set number of mines on the field. The mines would be placed randomly at the beginning, but adjust positions if necessary to prevent the user from losing immediately.
3	1	EECS 447 - Final Project	Write a query for a database system for a physical library. The query returns all overdue books and items for a specified client of the library. Includes error status codes for future frontends.
4	1	EECS 168 - Assignment 10	Implemented a simple DMV system that parses a records file, cleans the data, and prompts the user to search for a person in the system and displays their license information
5	2	EECS 168 - Assignment 7	Built a txt analyzer that removes extra whitespace and punctuation and tells a user how many times each word appears in the text and highlights which words are unique
6	2	EECS 168 - Assignment 8	Built a Python Pokémon battle builder that loads a Pokédex from file, translates English names, generates random teams, and simulates round-based battles with random outcomes, displaying winners and team states via a simple CLI.
7	2	EECS 268 - Assignment 4	Simulated a search history tracker within a browser application that navigates to pages, goes forward and backwards through pages, and shows search history using a linked list data structure.
8	2	EECS 268 - Assignment 3	Created a simulation of a CPU scheduler that takes an input file with formatted instructions, guided by rules of when and what instructions can be run at a certain point in the process.
9	3	EECS 330 - Lab 10	Created multiple advanced graph algorithms including Dijkstra's Shortest Path Tree Algorithm to find the shortest path in a weighted or directed graph, Prim's Minimum Spanning Tree, and Kruskal's Minimum Spanning Tree to find the minimum spanning tree of a connected and unidirectional graph
10	3	EECS 388 - Lab Final Project	Implement a path following algorithm on an embedded platform for a simple robotic car with LIDAR, visual data, and servo motors. Provides capabilities for steering to follow a given path through visual input, collision avoidance and fine tuning through LIDAR distance detection, and precise motor actuation and control.
11	3	External Project	Integrate with HTML geolocation and Google Maps APIs to provide an interface for users to see other users and points of interests within a nearby area. User indicators are dynamic and move around on the map.
12	5	EECS 210 - Assignment 8	Implemented multiple graph and game theory algorithms: Euler and Hamilton circuit detection using DFS, Dirac's and Ore's theorems, and a min-max Nim game simulation with 100 randomized trials. Demonstrates theorem application.
13	5	EECS 348 - Arithmetic Expression Evaluator	Developed an expression calculator in C++. The implementation parses infix input, converts it to postfix notation, and computes results. It supports operators (+, -, *, /, %, ^), handles parentheses, and ensures error-checking for invalid input and division by zero.
14	5	EECS 268 - Assignment 6	Implemented a recursive flood fill algorithm that simulates water spreading from a starting point across a map. It checks valid adjacent spaces, marks flooded areas, and stops when water runs out, adapting dynamically to any landscape layout.
15	5	EECS 510 - Final Project (Parser)	Developed a general purpose Pushdown Automata evaluator in Python: Allows user to define automata states and transitions, manages initial and accepting state conditions, manages PDA stack additions and removals, and outputs accepted status of input with the explored PDA path.
16	8	EECS 649 - Assignments 1 and 2	Assignment 1 and 2 involved implementing and analyzing regression and logistic regression models on real-world datasets in Google Colab, applying modeling techniques, evaluating performance, and submitting results via shared Colab links.

17	8	EECS 649 - Assignment 4	This assignment had two parts: build a basic RNN to predict the next values in a time series, and build an LSTM model to perform sentiment analysis in IMDB movie reviews. Both parts involved data preprocessing, building the model, training the model over multiple epochs, and evaluating its accuracy.
18	8	External Project	Develop an algorithm to discover optimal hotspots on a 2D heatmap with several trillions of cells. The algorithm generates the heatmap from reference functions and calculates the hotspots at once in realtime to maintain fast runtime and low memory overhead. Includes extensive configuration through a CLI and GUI.
19	8	EECS 678: Lab 8	Develop implementations for several CPU scheduling algorithms. Implemented schedulers included FCFS, Shortest Job First, Round Robin, Priority Queue, and preemptive versions of these algorithms. The implementation must exactly match a reference output for a set of scheduled processes.
20	13	External Project	Develop a server management platform for a small cluster of servers. The backend includes abilities to manage installed software, perform system updates, filesystem access, and service management. Included multiple frontends: a simplistic web interface, more extensive standalone desktop application, and a Discord-command-based bot frontend.
21	13	EECS 447 - Final Project	Developed an SQL-based library database system to manage books, borrowers, transactions, and overdue tracking. Included advanced queries, fines, and reporting features. Implemented using MySQL with structured documentation, team collaboration, and VC through GitHub.
22	13	EECS 678 - QUASH Project	Created a custom shell called Quash, a simulated command-line interface supporting common shell commands. Implemented with GNU Bison and Flex for robust parsing and error handling. Features include build via Makefile, cross-platform support, and thorough testing on multiple environments.