

Milestone 5 Progress Evaluation

Project Title: Improved Visualization for Formal Languages

Group Members: Chris Pinto-Font (cpintofont2021@my.fit.edu) , Vincent Borrelli (vborrelli2022@my.fit.edu), Andrew Bastien (abastien2021@my.fit.edu), Keegan McNear (kmcnear2022@my.fit.edu)

Faculty Advisor: Dr.Luginbuhl (dluginbuhl@fit.edu)

Client: Dr.Luginbuhl

Milestone #5 Progress:

Task	Completion %	Chris	Vincent	Andrew	Keegan	To Do
NFA dead state and bug fixing	80%	30%	30%	20%	20%	Add dead state creation
Canvas GUI upgrade with quality of life features	100%	30%	15%	40%	15%	Make further improvements as they arise and continue to rearrange buttons and GUI features
Tutorial mode	80%	25%	25%	15%	15%	Continue to bug test and stress test current system
String input converted into a DFA	80%	30%	30%	10%	10%	Continue to bug test and stress test current system

User feedback	80%	40%	20%	20%	20%	Continue to collect user input/judge input based on .exe release package
Update Read Me File	100%	30%	30%	20%	20%	Continue making updates as they arise
Poster Design	100%	25%	40%	25%	10%	N/A

Milestone #5 Discussion (Task Details):

- NFA dead state and bug fixing
 - Improve the robustness of our NFA feature integration via the inclusion of proper dead state creation during NFA conversion. We too continued to bug test our conversion process via manually testing several conversions for accuracy.
- Canvas GUI upgrade with quality of life features
 - Part and parcel of the refactorization process/continued expansion of features was the necessary step to rearrange the GUI layout to improve user experience, visual cohesion, and navigability. We moved the animation speed adjuster to the bottom of the screen to clear up room on the top bar for program features.
- Tutorial mode
 - Implemented an interactive, brief, tutorial mode into the program that shows the user how to operate the program via a demonstration popup segment.
- String input converted into a DFA
 - Implemented a feature within the program where in which the user can press a button then enter a collection of accepted strings to which the program will render out a DFA which meets the accepting criteria. Faced some issues in where the more accepted strings entered increases the algorithmic complexity.
- User feedback
 - Continued to collect user feedback through advisor correspondence, as well as publicly posting .exe file versions of rough cuts of the program for students to begin using the program.
- Update Read Me File
 - Continued to update the readme file as to expand scope of instructions for users to better understand the program's functionality and allow them to navigate the program more efficiently.
- Poster Design
 - Designed and illustrated a large scale poster displaying program features and

purposes as to present the finished product more effectively to an audience of investors and potential users.

Milestone #5 Discussion (Team Contribution):

- **Chris: Helped focus team goals, took charge on NFA design and dead state inclusion.**
- **Vincent: Worked on creating the solo executable program file, code inclusion, and bugfixing.**
- **Andrew: Assisted in code implementation and bug fixing**
- **Keegan: Assisted in code implementation and bug fixing, spearheaded further refactoring and GUI adjustments.**

Plan for Milestone #6:

Task	Chris	Vincent	Andrew	Keegan
Finalized Touches on user experience/bug testing	Bug Fixer and Project stress tester	Bug Fixer and Project stress tester	Bug Fixer and Project stress tester	Bug Fixer and Project stress tester
Zoom out/item selection and moving feature	Bug Fixer/Code Contributor and researcher	Bug Fixer/Code Contributor and researcher	Co-Lead coder and development head	Co-Lead coder and development head
Teaching mode	Bug Fixer/Code Contributor and researcher	Bug Fixer/Code Contributor and researcher	Co-Lead coder and development head	Co-Lead coder and development head
Update Poster as new features are added	Co-lead Writer	Co-lead Writer	Advisor	Advisor
Update “read me” file	Co-lead Writer	Co-lead Writer	Advisor	Advisor

Milestone #5 Plan Discussion (Task Details):

- Finalized Touches on user experience/bug testing
 - Put finishing touches on the final shippable version of the program, evaluate what needs to be finished for the program to exist in a workable state to the program as per project team standards.
- Zoom out/item selection and moving feature
 - Expand the scope of the program’s canvas space by allowing the user to zoom out and select canvas items to move and delete at will to improve user experience with aforementioned canvas graphing space.

- Teaching mode
 - Implement an interactive feature for the program that instructs the user on how to construct and traverse DFAs in a constructive manner.
- Update Poster as new features are added
 - Further expand the poster and readjust it to fit further features as the project concludes.
- Update “read me” file
 - Continue to update the onboard readme file as to include new features and be as instructional and robust as possible.

Dates of Meeting/Client Feedback:

August 27:

- Focus is on DFAs
- User should be able to create a DFA intuitively (we’d need to nail down what that meant)
- The system should be able to check for completeness (e.g., are there transitions missing?) or correctness (is it truly a DFA) - Maybe the system should be able to complete given an incomplete DFA (e.g., fill in missing transitions – but how?)
- Eventually, it would be nice to format/prettify the DFA graphs (align nodes, improve readability)
- Be able to designate initial and final states, and maybe even designate a state as a dead state, and let the system do its magic
- Multiple symbols on a single transition between arcs in a way that is natural and readable
- Animate processing of strings on a DFA
 - step-by-step, play, pause, etc.
- Minimization
- The vision is, given a set of strings that are accepted and not accepted, to let the system create a minimal DFA (in terms of states) that accepts/rejects as specified

September 10:

- Involve in the program is lambda transitions.
- The key things needed is when given strings it would create the DFA, when given strings, it would have to determine whether it accepts or rejects, and be able to create these states whenever

October 24:

- Discussed update on milestone two progress
- Expressed current work on internal program logic
- Spoke about current in-program node connecting format
- Expressed interest in further implementation of and use of the visual canvas type graphing form.
- Spoke to us about our current challenges regarding the state of the project and its code going forward.

November 10:

- Discussed the steps needed to continue the project and what should be done.

November 20:

- Discussed how minimization works by paper and the algorithm to use.

November 21:

- Discussed progress, which was a bit minimal.
- Talked briefly about minimization

January 20:

- Discussed project plan

February 20:

- Discussed what progress was made, which was the NFA logic and detection
- Figured out future steps to take

Faculty Advisor Signature: _____ **Date:** _____