MANTECH Publications

Journal of Research in Computer Science and Engineering
Volume 11, Issue 3, September-December 2026
ISSN: 2457-0818 (Online)

AlgoTrace: An Interactive Algorithm Visualization and Pattern-Based Coding Practice Platform

Shravani Patil[1], Kasturi Tomake[2], Samiksha Patil[3], Vaishanvi Patil[4]

ABSTRACT

Data Structures and Algorithms are parts of computer science and play a vital role in building software thinking logically solving problems and getting ready for technical interviews. Many students find it hard to understand how algorithms actually work when they only read theory or look at examples. Algorithm visualization gives a way to show how an algorithm runs in pictures helping learners see the actions taken and how data changes as the algorithm proceeds [1]. Studies on algorithm visualization say that the learning value of a tool depends on how learners interact with it and take part in the learning process [2] [3].

AlgoTrace is a platform that lets students watch algorithms step by step. The platform focuses on categories such as sorting, searching, graph, tree and greedy algorithms. Interactive buttons like Next Previous and Reset let users move through each step go back to steps and restart the demonstration. These features help students follow the run easily and link theory to real examples.

In addition to showing steps AlgoTrace aims to give a way to practice coding. A pattern-based learning module groups problems around techniques such as Two Pointers, Sliding Window, Arrays and Hashing, Binary Search, Stack, Dynamic Programming and Greedy methods. A company-based preparation module can further organize interview questions by company, difficulty and problem-solving pattern. These approaches help students spot recurring problem types and prepare for interviews in a more organized way. Existing resources such as VisuAlgo and LeetCode show ways to visualize algorithms and practice coding [4] [5].

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

The overall goal of AlgoTrace is to combine algorithm visualization and structured coding practice in one learning environment. The platform is meant to supplement classroom lessons by letting students observe algorithm execution understand each solving step and practice Data Structures and Algorithms systematically. Future development may include progress tracking, personalized learning recommendations, code-linked visualization and more practice resources. The usefulness of the platform should be checked through testing, usability assessment and learner feedback before claiming it improves learning outcomes.

Future plans could add progress tracking, personalized guidance, visual tools that link directly to code and extra practice material. The platforms impact should be measured with tests, usability reviews and learner comments before making claims about its effect, on learning.

KEYWORDS: *AlgoTrace, Data Structures and Algorithms Algorithm Visualization, Interactive Learning, Coding Practice, Problem-Solving Patterns, Technical Interview Preparation.*

INTRODUCTION

1. Background Data

Structures and Algorithms or DSA are parts of computer science. Data structures help. Store data. Algorithms tell us how to process that data and solve problems. Things like arrays, linked lists, stacks, queues, trees, graphs, searching, sorting and greedy methods are common in programming. Knowing these helps students think logically write code and break down tough problems step by step.

Dsa can be hard for new learners. A student might know what an algorithm is. Still not see how it works. Take sorting algorithms for example. You need to see how items are compared how they swap places and when the sorting ends. Graph algorithms need you to follow how nodes are visited how edges connect them and how distances or paths change. Just reading about these or looking at pictures often doesn’t show the whole process clearly.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

That's where algorithm visualization helps. It shows how an algorithm runs, step by step in a way. Studies have looked at how useful these visual toolsre for learning [1]. They also say learners should interact with the visuals not just watch them [2]. This gives a starting point for building tools that let students explore algorithms at their own pace.

2. Need for an Interactive Learning Platform

Students usually learn DSA from class notes, textbooks, videos coding websites and online platforms. Each helps in a way.. Often they have to jump between many sources to understand an algorithm watch it run, practice problems and get ready for interviews. This can be messy for beginners who are still building their problem-solving skills.

An interactive platform can fix this. Of only reading or watching a full animation learners can control the algorithm's steps. They can pause, go back and focus on what matters. This helps them think through each move and see how one step leads to the next. The platform should explain things clearly have controls and not overload students with too much at once.

Research by Hundhausen, Douglas and Stasko looked at studies on algorithm visualization. Said we need to pay attention to how it's designed and how we test its value [1]. Naps and others also stressed the need for learners to actively engage with tools [2]. These studies show that visualization is a tool, not a fix. It works best when designed well and tested properly.

3. Proposed AlgoTrace Platform

AlgoTrace is a platform to help students learn DSA with clear visual demos. It links theory with algorithm steps. Algorithms are grouped into categories: sorting, searching, graph algorithms, tree algorithms and greedy algorithms. This helps learners find topics and explore based on what they need.

One main feature is step-by-step control using **Next, Previous and Reset** buttons. With Next a student moves to the step. Previous lets them go back. Reset starts over. These controls help with self-paced learning. For example while learning a sorting algorithm a student can watch two elements being compared see a swap happen and track how the array changes over time.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

The platform also explains each step by linking the changes to the logic behind them. Later it may highlight the line of pseudocode show important variable values and explain why a certain action happens. These should be added for each algorithm. Checked to make sure they match the actual logic.

4. Pattern-Wise Coding Practice

Knowing algorithms is only part of learning DSA. Students must also learn to spot the technique for a new problem. Two problems may look different. Use the same approach. Spotting that can help students reuse what they've learned.

To help with this AlgoTrace will include a pattern- coding practice section. Planned categories are Arrays and Hashing, Two Pointers, Sliding Window, Binary Search, Stack, Dynamic Programming and Greedy algorithms. Each category will have practice questions, explanations, difficulty levels and the method used. For example Two Pointers could include problems where two indices move through an array. Sliding Window might cover problems that look at a part of an array or string.

This pattern-based setup helps students go beyond memorizing solutions. First they learn the pattern then they can watch demos if available and finally they practice problems using that same technique. This part is still. Will depend on what gets built in the final version.

5. Company-Wise Interview Preparation

Technical interviews often ask candidates to solve coding problems under time pressure and explain their thinking. Students preparing for jobs need a way to practice. Platforms like LeetCode offer problems to practice on [5]. AlgoTrace wants to help by organizing interview questions into companypattern-wise groups.

The company-wise section will group questions by target companies, difficulty and DSA topics. It will also show which pattern a question uses so students can connect interview prep with what they've learned. This helps them plan practice and review topics. The questions will come from public sources and won't promise anything will appear in a real interview.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

6. Objectives and Scope

The main goal of AlgoTrace is to create a learning space that mixes interactive visualizations with organized DSA practice. Its goals are:

a) Show selected algorithms with step-by-step execution.

b) Give Next Previous and Reset buttons to move through steps.

c) Group algorithms into categories like sorting, searching, graphs, trees and greedy.

d) Help students understand algorithm actions with visuals and explanations.

e) Build a pattern- practice section for common problem-solving methods.

f) Try out company- question grouping for interview prep.

g) Plan improvements like tracking progress and suggesting personalized practice.

What's done should be separate from what's planned. Only features that are built and tested should be called part of the system. Pattern-wise practice, company- prep, progress tracking and personal recommendations will be listed based on what's actually built.

In short AlgoTrace wants to link learning with real coding practice. With demos and a clear path to follow it aims to be a useful tool for students learning DSA and getting ready, for programming interviews. Its success should be checked through testing, usability reviews and student feedback.

LITERATURE REVIEW

1. Algorithm Visualization in Computer Science Education

Algorithm visualization is a method used to show the actions of an algorithm using pictures moving images and changes in data. It can help students see how an algorithm works of just reading about it. For example a sorting algorithm can be shown using bars or array parts that move when they are compared and swapped. Also graph algorithms can be shown by highlighting the parts that are visited the edges that are chosen and how the path lengths change. Hundhausen, Douglas and Stasko did a study looking at research about how useful algorithm visualization's [1]. They looked at how visualization was used and how its effect on learning was measured. This research is important for making tools because it shows that the value of visualization should be checked through proper learning tasks and testing. Just having an animation does not mean students understand the algorithm.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

The results give a base for AlgoTrace. The platform uses step-by-step steps to make it easier to see how an algorithm works. Of showing only the final result the idea is to show the steps so students can follow each action.. Whether this design helps a certain group of students is something that needs to be tested separately.

2. Student Interaction and Algorithm Animation

Stasko looked at how students making their algorithm animations can be used as a way to learn [3]. This study looked at a method where students create animations of just watching ones made by others. This method shows how important it is to have students involved in showing and explaining algorithms.

Naps and other researchers studied how visualization and student engagement are connected in computer science education [2]. They said that visualization should be used with learning tasks and discussed how to check if learning happens. This is important because students might watch an animation but not know why something happens or how the algorithm gets the result.

These studies are important for how AlgoTrace's designed. The Next Previous and Reset buttons are meant to let students move through the algorithm steps go back to parts and start over. When there are visual cues and explanations these buttons can help students look at each step at their own speed. But the real benefits of these buttons need to be tested through user testing not just assumed.

3. Integration of Visualization with Algorithm Explanations

A point in algorithm visualization is connecting the visual part with the steps of the algorithm. A picture might show an array changing. Students might still need an explanation of why the change happened like a comparison, a condition, a variable change or a decision. So visualization and explanations should work together.

AlgoTrace wants to solve this by breaking algorithm demos into steps that're easy to understand. For some algorithms future work can link each part to the exact line of pseudocode the variable values and a short explanation of what is happening. For example a sorting demo might show which two elements are being compared and if a swap is needed. A graph algorithm might explain why a vertex is chosen or why a distance value changes.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

This way of working follows the idea that algorithm visualization should help students engage and understand [1] [2]. However features like code highlighting looking at variables and explanations in language should only be added when they are available and checked in the real system.

4. Existing Algorithm Visualization Resources

VisuAlgo is a website that shows data structures and algorithms with explanations and animations [4]. It shows how visuals can be used to explore how algorithms and data structures work. Such resources are useful when comparing the part of AlgoTrace.

Existing tools show how important it is to show how an algorithm works so students can see it.. A learning platform may also need to think about how students move from understanding an algorithm to using it on their own. A student who understands a sorting animation might still need to practice to find the approach for a new problem.

AlgoTrace is meant to link algorithm visualization with an organized learning path. Its categories help students find algorithms and its interactive buttons help them go through steps. The plan is to add pattern-based practice and interview prep resources so learning goes beyond visuals. This is a goal for AlgoTrace, not a claim that other tools don’t have these features

5. Online Coding Practice and Problem-Solving Patterns

Online coding sites let students practice coding problems, test solutions and get better at solving problems. LeetCode is a site used for coding practice and interview prep [5]. These sites let students work on problems that need knowledge of algorithms, data structures and coding languages.

New students might find it hard to know which method to use for a specific problem. Remembering answers doesn’t always help them see the pattern in a new problem. So organizing practice questions by problem-solving methods can be a way to learn.

AlgoTrace wants to add a pattern-based practice section with categories like Two Pointers, Sliding Window, Arrays and Hashing, Binary Search, Stack, Dynamic Programming and Greedy algorithms. Each category can have questions, explanations and difficulty levels. This way students can see how similar problems are and practice one method with examples.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

This section is planned for the platform unless it is already there. How well it works can be checked by seeing if students can spot the pattern and solve similar problems after doing the learning activities.

6. Company-Wise Interview Preparation

Preparing for interviews is another part of learning DSA. Candidates often need to review algorithms and practice problems that match the questions in interviews. Online coding sites have problems but students might also have their own lists of questions for specific companies.

AlgoTrace wants to organize interview prep questions by company, difficulty and problem-solving pattern. The goal is to make practice more structured and help students link their DSA learning to their career goals. For trust the questions should come from sources and company names should be supported by public evidence. The platform should not say a question will definitely be in an interview.

7. Research Gap and Motivation for AlgoTrace

The studies reviewed show points for tools that help students learn algorithms: showing the visuals keeping students involved and testing the results [1]–[3]. Existing resources also show how useful algorithm visualization and coding practice are [4] [5]. These resources form the basis for AlgoTrace.

The reason for AlgoTrace is to bring step-by-step algorithm visualization and structured coding practice into one learning place. The platform wants to help students move from watching how an algorithm works to practicing problem-solving methods and getting ready for interviews. The plan includes organizing questions by company tracking progress and giving suggestions.

So the main idea of AlgoTrace is to bring these learning parts not to say that visualization or coding practice is new. More work is needed to check the accuracy of the visuals how easy the interface is, the quality of the learning content and how helpful the platform is for students. A proper test, with tasks and user feedback will show how well AlgoTrace meets its learning goals.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

FUTURE SCOPE AND RESEARCH CHALLENGES

Although algorithm visualization platforms help students understand Data Structures and Algorithms there is still room to make the learning experience better. I think students would benefit from a personalized approach. One possible direction is to introduce learning recommendations that are based on students performance, completed activities and areas of difficulty. By analyzing practice history the system could suggest algorithms and coding problems that match each learners progress. This may help students focus on topics and follow a more structured learning path. Another important direction is to improve the connection between algorithm visualization and practical coding. Students may understand the execution of an algorithm but still find it difficult to implement the same logic in a programming language. Integrating code explanations with step-by-step visualization could help learners understand how individual statements affect the algorithms execution.

Supporting programming languages could further improve the usefulness of the platform for students with different learning preferences. AlgoTrace could also be extended through patternbased coding practice that covers techniques such as Two Pointers, Sliding Window, Arrays and Hashing, Binary Search, Stack, Dynamic Programming and Greedy algorithms. Organizing problems by difficulty level and problem-solving pattern may help students recognize approaches and apply them to unfamiliar problems. Company-wise preparation modules could also organize coding questions according to students interview preparation goals. Such collections would require verification and updates to maintain their relevance and accuracy. The introduction of learning assistance is another possible area for future development. An AI-based assistant could provide hints explain algorithm steps and guide learners toward identifying errors without revealing the complete solution. However the reliability of generated explanations the accuracy of suggested solutions and the need to avoid encouraging dependence on automated assistance would require evaluation. Progress tracking could also be improved by providing analytics including topic-wise completion, practice consistency and frequently encountered difficulties. These insights could help learners identify areas that require attention. Future versions could also support account-based synchronization so that learning progress remains available across multiple devices.

Despite these opportunities several research challenges remain, including maintaining visualizations supporting a growing collection of algorithms ensuring a consistent user

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

experience and evaluating whether interactive learning improves students understanding. Additional considerations include accessibility, application performance, data privacy and the reliability of learning resources. Future research could investigate these factors through usability testing, learner feedback and comparisons of learning outcomes. Overall the future development of AlgoTrace can focus on learning, code-linked visualization, intelligent assistance, structured coding practice and improved progress tracking. These enhancements aim to create an integrated learning environment that supports students from understanding fundamental algorithms to developing problem-solving skills, for technical interviews.

RESULTS AND DISCUSSION

1. How Well Was the System Designed and Implemented?

AlgoTrace is built as a learning platform that brings together algorithm visualization and structured coding practice. The goal is to help students grasp how algorithms work by showing them step by step. Visual representations let learners see what happens during each operation such as sorting or searching. Controls like Next Previous and Reset allow students to go forward go back or restart execution whenever needed. The platform groups algorithms into categories—sorting, searching, graphs, trees and greedy algorithms. This makes it easier for learners to find what they need and follow a path through different topics. The design focuses on clarity and user control aiming to improve understanding and reduce confusion.

2. How Does the System Support Algorithm Learning?

AlgoTrace supports learning by using simulations of algorithm execution instead of relying only on text-based explanations. When a student runs a visualization they can watch how data changes over time—how elements are compared, moved or traversed. This hands-on view helps learners see the logic behind each step and understand the flow of an algorithm. For example watching bubble sort swap elements or seeing how binary search narrows down the search space can make abstract ideas more concrete. The impact of this approach on learning should be tested through student use. We need to know if it actually helps students get better at understanding data structures and algorithms especially if they struggle with the theory.

3. Is the Platform Easy to Use?

Usability is a part of the AlgoTrace design. The platform organizes algorithms into labeled categories so students can find what they are looking for quickly. Navigation is kept simple

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

with buttons and a clear layout. The controls are straightforward meaning learners don't need to spend time figuring out how to use the interface. Instead they can focus on the algorithm itself. Still to be sure the platform is user-friendly we should test it with students. Observing how students move through the system can reveal issues with navigation or unclear steps. Feedback from both beginners and advanced users will help improve how algorithm steps are shown and whether the interface works well for different types of learners.

4. How Does AlgoTrace Support Coding Practice and Interview Preparation?

Beyond visualization AlgoTrace gives students tools for coding practice and interview prep. It includes problem-solving patterns such as Two Pointers, Sliding Window, Arrays and Hashing, Binary Search, Dynamic Programming and Greedy algorithms. These patterns help learners recognize techniques and apply them across different problems. Questions are grouped by difficulty—medium, hard—and by technique. This helps students build skills step by step. There are also collections of questions from companies so students can prepare for technical interviews based on their job goals. Progress tracking features let users see what they've completed and which topics they still need to work on. This structure supports both learning and real-world preparation.

5. What Are the Limitations and Future Improvements?

While AlgoTrace offers a lot it still has room to grow. The platform's value depends on how accurate the visualizationsre how clear the explanations are and how strong the practice questions are. Keeping the algorithm library. Maintaining up-to-date company interview questions will require ongoing effort. Future versions could include personalization, such as recommended topics based on a user's learning history. Code-linked explanations could show how the code matches each step in the visualization. Intelligent hints might help students when they get stuck. Detailed analytics could help users understand their progress and weaknesses. Testing with students will be essential to see how effective the visuals are, how usable the interface is and how well the platform supports learning.

In summary AlgoTrace tries to bridge the gap between theory and practice. It combines learning, hands-on coding and interview prep into one system. By offering visuals, structured practice and progress tracking it aims to make learning data structures and algorithms more organized and effective. To measure how well it works, continued evaluation and real user

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

feedback are necessary. This feedback will guide updates and help improve the platform, for all types of learners.

CONCLUSION

Learning Data Structures and Algorithms (DSA) can be difficult for students. Understanding the theory and using it to solve coding problems takes practice and strong logical thinking. AlgoTrace is built as a learning platform that brings together algorithm visualization and structured coding exercises. It is meant to help students at every stage of their DSA journey. The platform makes it easier to understand how algorithms work by showing step-by-step representations of the execution process.

AlgoTrace focuses on algorithm topics such as sorting, searching, graph algorithms, tree algorithms and greedy algorithms. Learners can control the flow of the algorithm using tools like Next, Previous and Reset. These controls let students slow down go back or restart the process to see how data changes at each step. This helps them build an understanding of the logic behind each algorithm.

The platform also helps with coding practice by organizing problems into categories. It includes problem-solving patterns and modules based on companies that often ask DSA questions. This helps students prepare for interviews in a more organized and efficient way. Students don’t have to guess what to study. They can follow a path that matches real-world interview patterns.

A strong learning environment is at the heart of AlgoTrace. The platform connects concepts with real coding tasks helping students see how ideas apply in practice. Features like progress tracking let learners see how far they've come. They can check which topics they’ve completed and which ones need work. This keeps students motivated and consistent in their learning.

By combining visualization, structured practice and progress monitoring AlgoTrace reduces the need to jump between websites or tools. Students can learn, practice and prepare in one place. This makes the learning process smoother and more effective.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

In the end AlgoTrace aims to offer a accessible and well-organized space for students to grow in DSA. It supports coding practice and technical interview prep in one place. Future updates could include learning suggestions, code-linked explanations, smart hints and better learning analytics. These improvements could help students learn faster and more effectively.

To see how well the platform works it's important to test it with students. Usability, reliability and impact on learning outcomes all need to be studied. Feedback from students will help improve the experience. Over time AlgoTrace can become a tool, for anyone learning Data Structures and Algorithms.

REFERENCES

1. C. D. Hundhausen, S. A. Douglas, and J. T. Stasko, "A Meta-Study of Algorithm Visualization Effectiveness," Journal of Visual Languages & Computing, vol. 13, no. 3, pp. 259–290, 2002. doi: 10.1006/jvlc.2002.0237.
2. T. L. Naps et al., "Exploring the Role of Visualization and Engagement in Computer Science Education," ACM SIGCSE Bulletin, vol. 35, no. 2, pp. 131–152, 2003. doi: 10.1145/782941.782998.
3. J. T. Stasko, "Using Student-Built Algorithm Animations as Learning Aids," in Proceedings of the 28th ACM SIGCSE Technical Symposium on Computer Science Education, 1997, pp. 25–29. doi: 10.1145/268084.268091.
4. VisuAlgo, "Visualising Data Structures and Algorithms Through Animation." [Online]. Available: https://visualgo.net/
5. LeetCode, "LeetCode: The World's Leading Online Programming Learning Platform." [Online]. Available: https://leetcode.com/

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved

***Author for Correspondence**

Shravani Patil

E-mail: pshravani1276@gmail.com

[1,2,3,4]Student, Department of Computer Science and Business Systems, Kolahpur Institute of Technology's College of Engineering, Gokul, Shirgoan, Kolahpur

Received Date: October 14, 2026

Accepted Date: October 18, 2026

Published Date: October 22, 2026

Citation: Shravani Patil, Kasturi Tomake, Samiksha Patil, Vaishanvi Patil**.** AlgoTrace: An Interactive Algorithm Visualization and Pattern-Based Coding Practice Platform. Journal of Research in Computer Science and Engineering. 2026; 11(3): 133-146p.

Page 133-146 © MANTECH PUBLICATIONS 2026. All Rights Reserved