



## **A Visual and Engaging Approach to Learning Computer Algorithms**

**Dr. Daniel Raviv, Florida Atlantic University**

**Yumi Kahori Nakagawa**

**Mr. George Roskovich**

# **A Visual and Engaging Approach to Learning Computer Algorithms**

**Daniel Raviv, George Roskovich and Yumi Nakagawa**

**Department of Computer & Electrical Engineering and Computer Science**

**Florida Atlantic University**

**Boca Raton, Florida 33431**

**Tel: (561) 297-2773**

**Emails: [ravivd@fau.edu](mailto:ravivd@fau.edu) , [groskovi@my.fau.edu](mailto:groskovi@my.fau.edu) and [ynakagaw@fau.edu](mailto:ynakagaw@fau.edu)**

## **Abstract**

Despite attempts by faculty to teach in ways that accommodate new styles of learning, there is much room for improvement. In order to adapt to students' continually changing learning styles, efforts must be made to further modify teaching methods that include more relevance in visual, intuitive, and interactive ways. This paper focuses on a "mini" experimental reform aimed at introducing difficult concepts in computer algorithms in a way that students can relate to. The method is based on establishing students' intuition by providing visual relevance-based content before focusing on mathematical understanding. The goal is to help students develop a core understanding of the subject matter, leading to an easier transition to deeper mathematical analysis.

This is part of a greater effort at Florida Atlantic University to apply this method to different subjects in engineering such as control systems, calculus, and MATLAB. To gauge the receptiveness of the methodology, the techniques were applied over the course of a semester for a class titled "Design and Analysis of Computer Algorithms". The results, although preliminary, have been positive. A larger effort is presently being conducted re-assess the success of the method by monitoring the progress of a class and its individuals as the semester moves on.

## 1. Introduction

According to research, 65% percent of the population benefits more from visual learning than any other style<sup>1</sup>. Incidentally, 90% of the information that the brain processes is visual<sup>2</sup>. Although there is a lot of research to conduct regarding visualization as a tool for teaching and learning, there is a consensus in terms of the effectiveness of catering to this style of learning. Studies have shown, students with different learning styles favor visualization as a means of learning a new concept<sup>3</sup>. Considering those facts, the learning examples for this approach to teaching computer algorithms strives to use the different visualization tools and engaging material in order motivate students to understand the concepts abstractly as an introduction to pre-existing materials. It is also worth mentioning that different disciplines might require different approaches to teaching. Thus, many educators are using more progressive education methods and techniques to help their students learning process such as collaborative teaching (students and professors learn from each other), and Blend Learning (using technology and traditional education methods), just to name a few.

This paper presents an approach for teaching and learning computer algorithms in an engaging, visual, and interactive way. Traditionally by observing algorithm books is clear that concepts are being taught in mostly a mathematical way, skipping visualization. Concepts such as Binary Search, Recursion, and Sorting are presented in a mathematical way, many times without a single real life example. We introduce these same ideas in different ways, appealing to students new, and emerging learning styles.

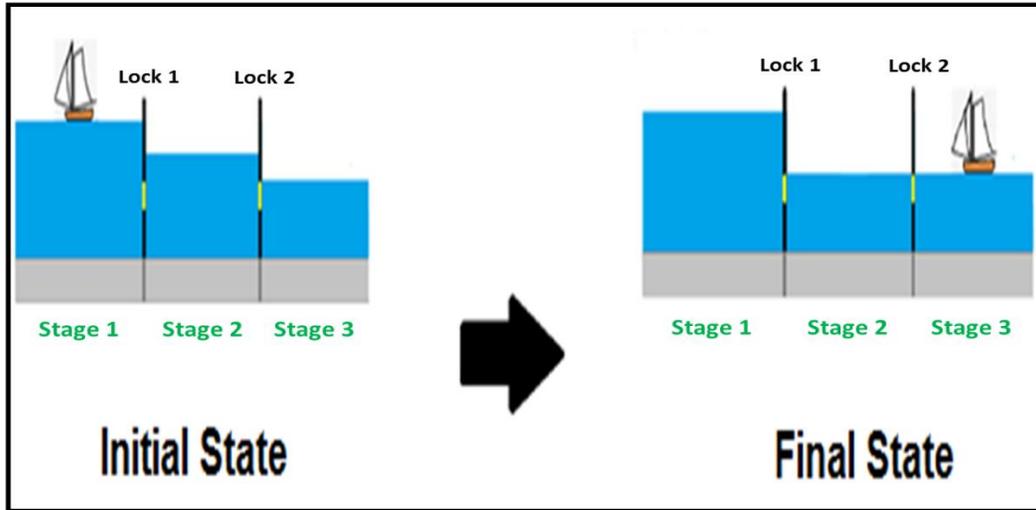
## 2. Methodology

This section provides examples of the methods used in the manuscript to teach different topics covered in computer algorithms in a way that caters to visual styles of learning. The goal is to establish students' intuition by providing them with an abstract view of material through visual and/or engaging examples. These examples would typically precede traditional content for each respective topic in algorithms.

The first type of example utilizes a story based approach. It starts off with a process with which most people are familiar. Once the steps of the process are complete, a flow chart is displayed describing the steps that were made. Other methods include activity based exercises and puzzlers to keep students engaged.

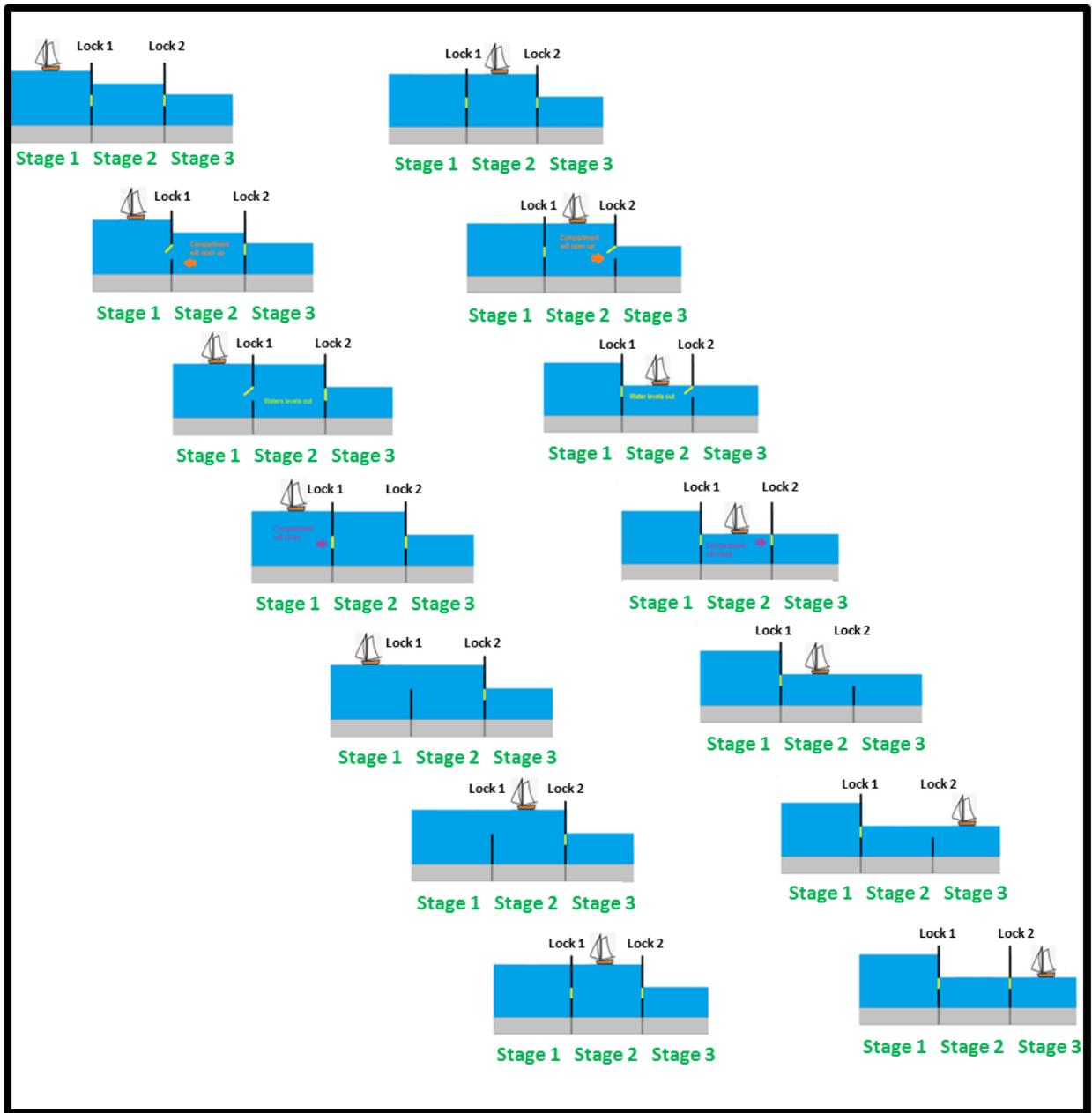
### **Basics of an Algorithm: The Panama Canal Example (*Story-based Approach*)**

The lock gate system of the Panama Canal is an example used to display the flow of an algorithm. The students are first introduced to the initial state of the ship and the final state as illustrated in figure 2.1 below.



**Figure 2.1** Panama Canal Start and End

Once the states are established, illustrations are provided along with an explanation of each step of the process. The various steps are all presented in figure 2.2 below. They show the steps necessary for a ship to successfully travel through the canal. First a compartment must open up, filling up the next stage. Once the next stage is filled, the gate must be lowered to allow the ship to enter. The process is repeated until the ship arrives in stage 3.



**Figure 2.2** Panama Canal Solution

The steps of the problem and the solution are followed by a comparison with its corresponding flowchart. This connects the understanding of the scenario with an algorithm. It also reinforces the use of flowcharts.

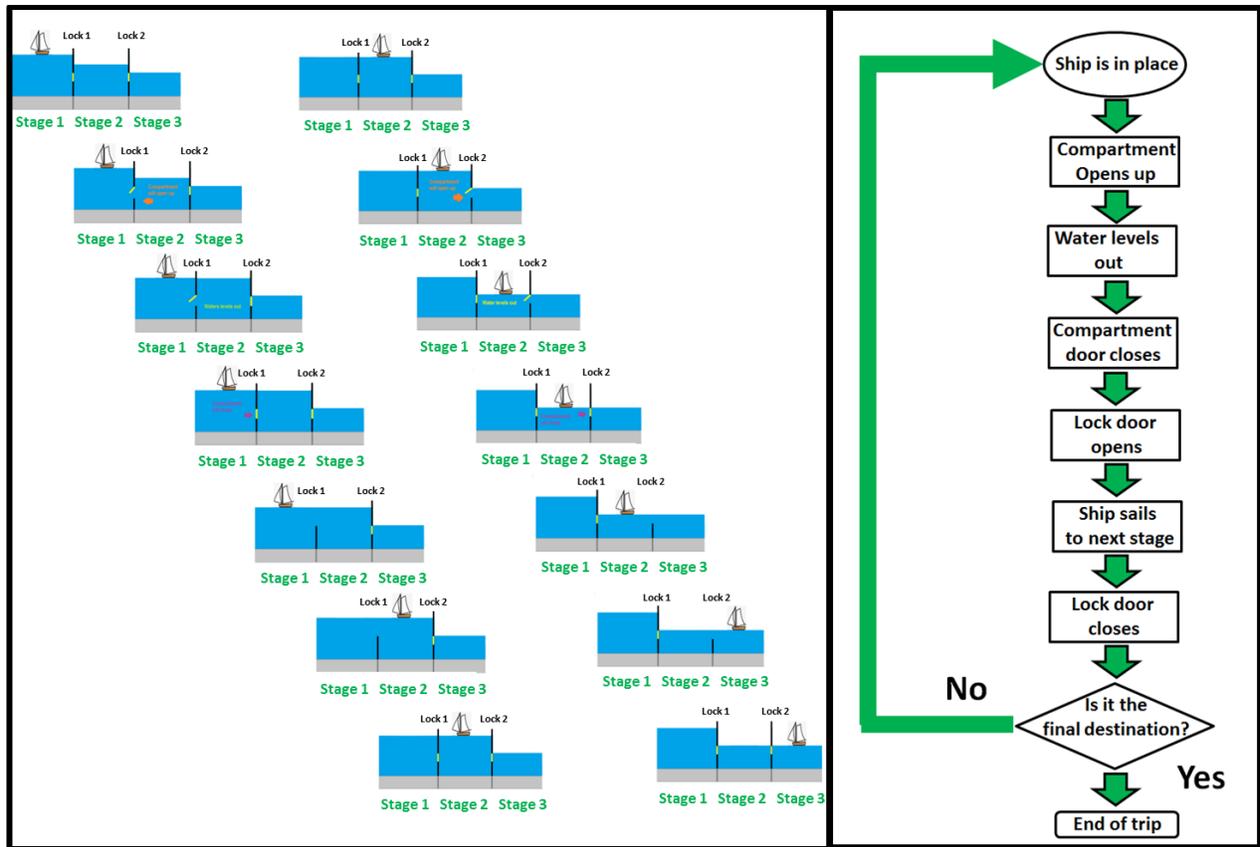
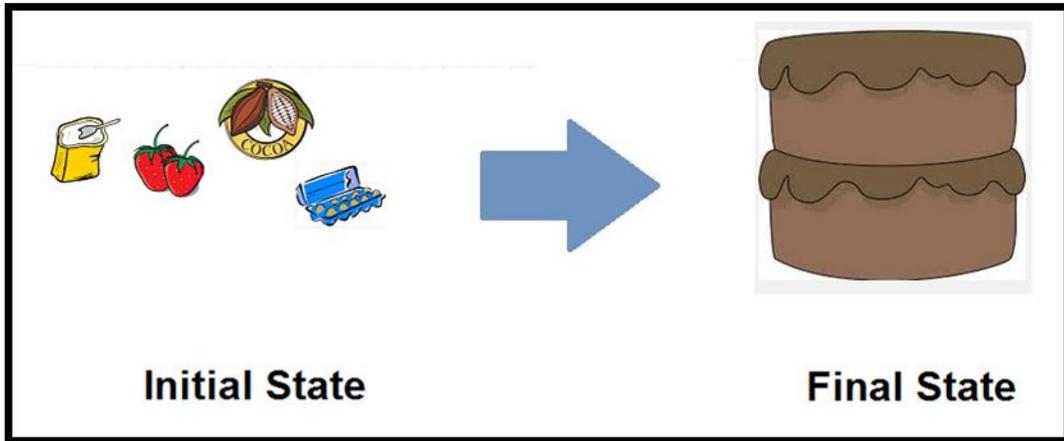


Figure 2.3 Panama Canal Solution and Flowchart

As an add-on, “Fun Facts”, “Did you know?” and “Just for Fun” sections are introduced as well. The purpose is to provide students a break while still approaching the topic at hand in simple terms. This ranges from trivia information to deeper and more mathematical information about a subject. The goal is to keep the reader’s attention by changing the pace, engaging them with fun, thought provoking questions.

**Basics of an Algorithm: Let’s Make a Cake Example (Story-based Approach)**

The following example intends to show the concept of an algorithm in a visual manner. The story is about a girl Martha, who wants to make a special cake for her mother with an old family recipe. The table in figure 2.5 illustrates the step-by-step instructions of how the cake is made. Before starting the process, the students are provided with a preview of the initial and final states.



**Figure 3.4** Cake Initial and Final State

Then, step-by step instructions are given.

### Choco Cake Recipe

1. Get the ingredients
  - Cocoa and Strawberries
  - 2 Eggs
  - 1 tablespoon of Baking Soda
2. Mix the ingredients
3. Pour mixture in the pan
4. Bake
5. Test it with a fork to check if the cake is ready. Keep baking it if it is not ready
6. Remove from oven
7. Let it cool down

**EAT IT!**

**Figure 2.5** Cake Recipe Instructions

# CHOCO CAKE RECIPE

1. Get the ingredients

*Cocoa and Strawberries*

*2 Eggs*

*1 tablespoon of Baking Soda*



2. Mix the ingredients



3. Pour the mixture in a pan



4. Bake



5. Test it with a fork to check if the cake is ready. Keep baking if it is not ready



6. Remove from oven



7. Let it cool down

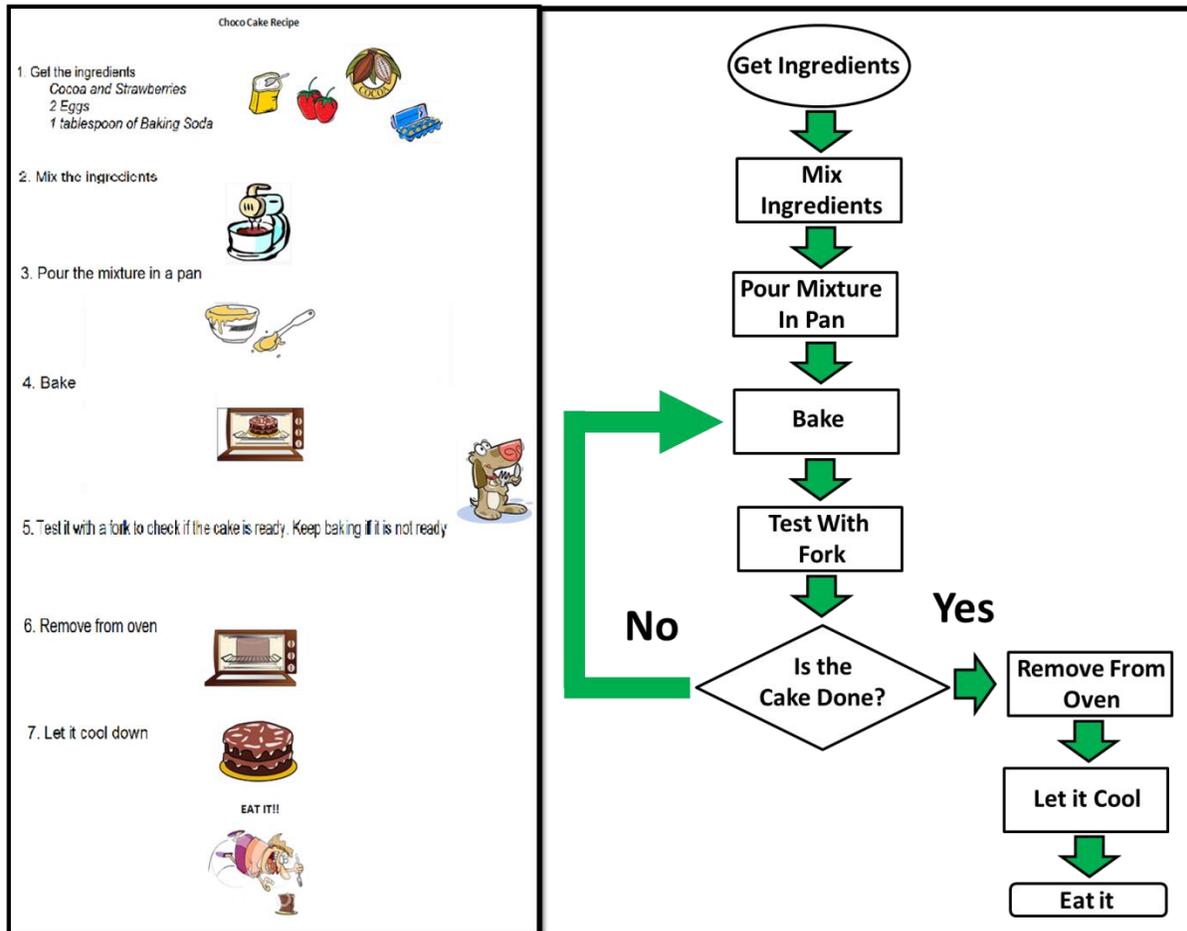


## EAT IT!



Figure 2.6 Cake Recipe Instructions with Visuals

After being provided with a visual step-by-step process, the flow chart is then displayed along side it to help students visualize the steps. In this manner, the student can visualize how the problem fits with the concept of how an algorithm works.



**Figure 2.7** Cake Recipe Instructions with Visuals and Flowchart

**Activity: Dollar Bill Origami (Activity-Based Approach)**

This activity was based on the *Money Origami* book by Michael G. LaFosse<sup>4</sup>.



**Figure 2.8** *Money Origami* Book

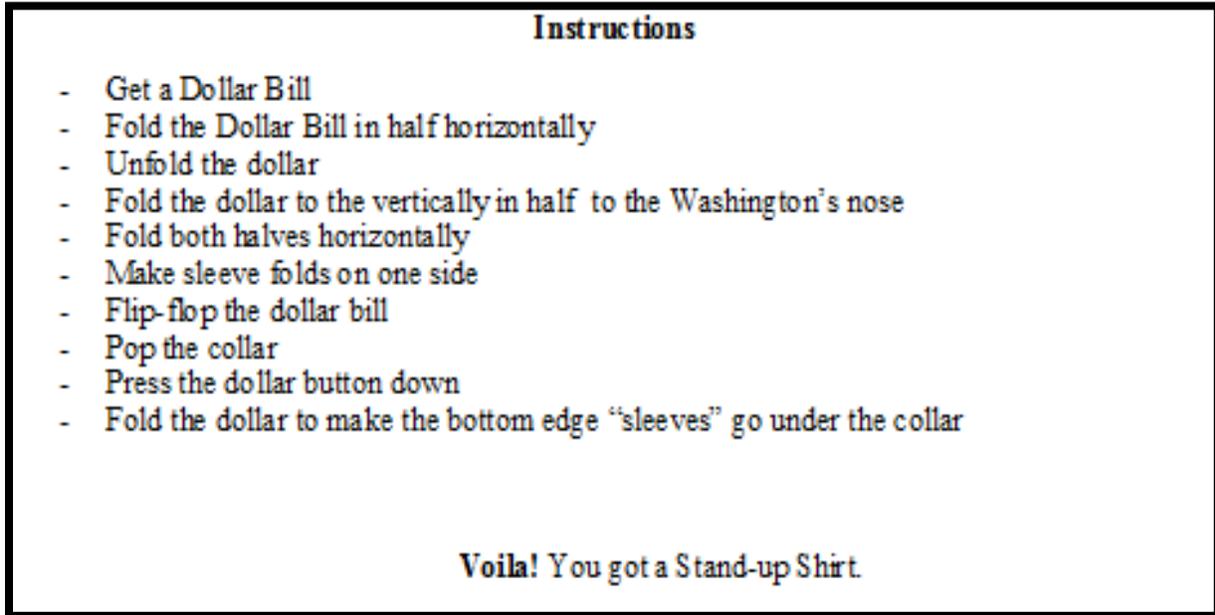
The goal of the activity is to teach students the meaning of what an algorithm is by making a stand-up shirt from a dollar bill. The activity can be done as a group or individual activity. The student is given instructions on how to perform the activity.

Origami is the art of folding paper and is a long standing tradition in Japan. Many people have modified traditional origami and have created innovative ways to practice this art. This activity will require a dollar bill instead of the traditional square origami paper. The goal is to make a stand-up shirt with a one dollar bill by following the step-by-step instructions. First, the initial and final states are shown.



**Figure 2.9** Origami Start and End

The instructions of the problem are provided in both text and visual form.



**Figure 2.10** Dollar Bill Stand Up Shirt Origami Instructions

## Instructions

1. Get a Dollar Bill



2. Fold the Dollar Bill in half horizontally



3. Unfold the dollar



4. Fold the dollar vertically in half to the Washington's nose



5. Fold both halves horizontally



6. Make sleeve folds on one side



7. Flip-flop the dollar bill



8. Pop the collar



9. Press the dollar button down



10. Fold the dollar to make the bottom edge "sleeves" go under the collar



Voila! You got a Stand Up Shirt

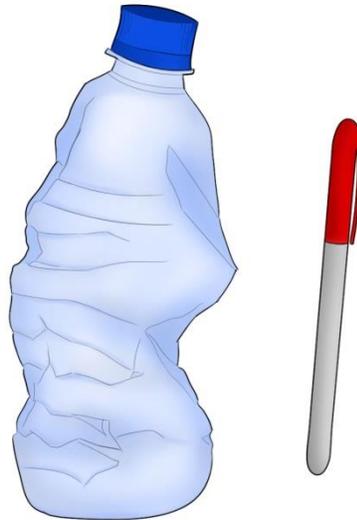


Figure 2.11 Dollar Bill Stand Up Shirt Origami Instructions and Visuals

The following exercise is also activity based. It is a puzzle based problem designed to get students to think and problem solve while enforcing the concept of recursion.

**Concept of Recursion: How to Survive in the Desert (*Activity-Based Approach*)**

Consider the following scenario. You are with your best friend in the desert and have only one bottle of water left. It's an oddly shaped bottle, no longer perfect and round but crushed and jagged from being in a backpack during your travels.



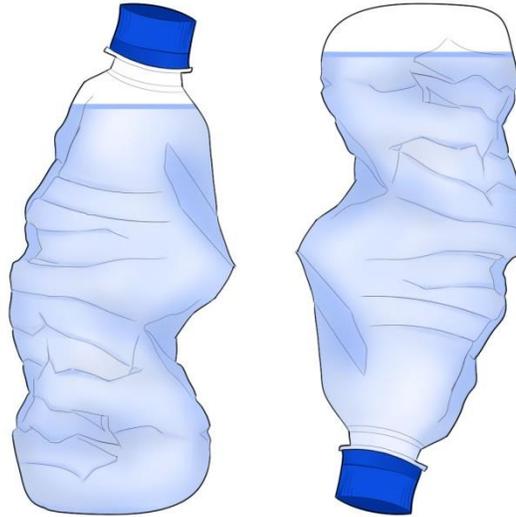
**Figure 2.12** Bottle and Marker

You must divide the water 50/50 for both you and your friend to survive but your cup is flat and cannot be used as a measurement device. You do, however, have a permanent marker.

How can you divide the water?

### Need a Hint?

There is little airspace at the top of the water bottle. What happens to the air when you change the direction of the bottle by flipping it upside down? By looking at the figure, what do you think is 50/50 now? How do you know?



**Figure 2.13** This is what it might look like.

How can you use this new knowledge? How do you think you could incorporate the marker to help you divide the water? Jot some ideas down here, or even grab a water bottle and pen and try the exercise in real life.

## How to survive in the desert!

You know that the airspace always floats to the part of the bottle facing up, whether that is the cap or the bottom of the bottle, and the amount of air is the same even if it looks different due to the bottle's shape. To divide the water 50/50, use this knowledge and your pen. In this solution, one of you will drink half first, then the other will finish the rest of the water.

- (1) Mark the airspace, then flip the bottle over and mark the airspace at the other end of the bottle.
- (2) Drink a little water, put the cap back on, and mark the airspace right-side-up and upside-down.
- (3) Keep drinking and marking until the airspace lines are in the same place.
- (4) When the lines meet, you know that the volume of the water is the same as the volume of the air in the bottle. and it is safe for your friend to drink the rest of the water.

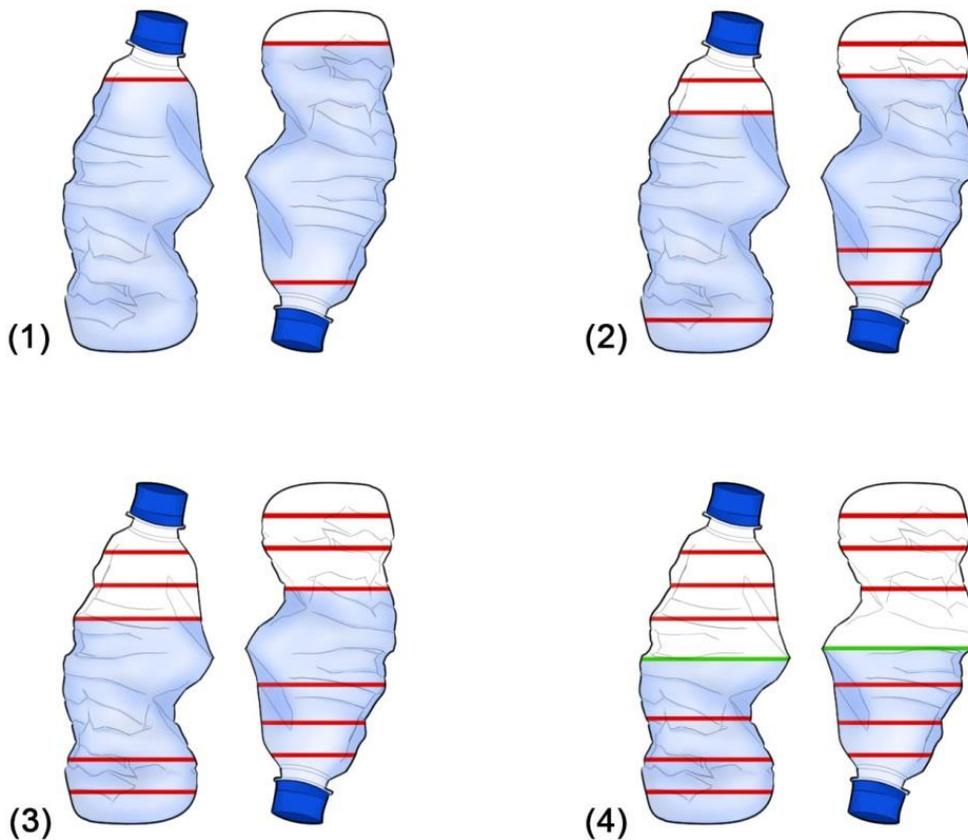


Figure 2.14 Water Bottle Solution

## Concept of Binary Search: Holy Grail Story (Story based Approach)

The Holy Grail story is used to explain and introduce the concept of binary search. The story is about King Arthur in search of the Holy Grail. He is told that the grail is in a wall guarded by monks who live in the wall's towers. Each monk has a name that represents each letter in the alphabet. They are also stationed in alphabetical order. In addition, the grail resides with the monk whose name starts with the letter "o". Remembering the teachings of his mentor Merlin, King Arthur decides to use binary search to maximize efficiency. The process to obtain the Grail is illustrated and explained step-by-step. This way the student is provided with a visualization of the method and gains an intuitive reference.

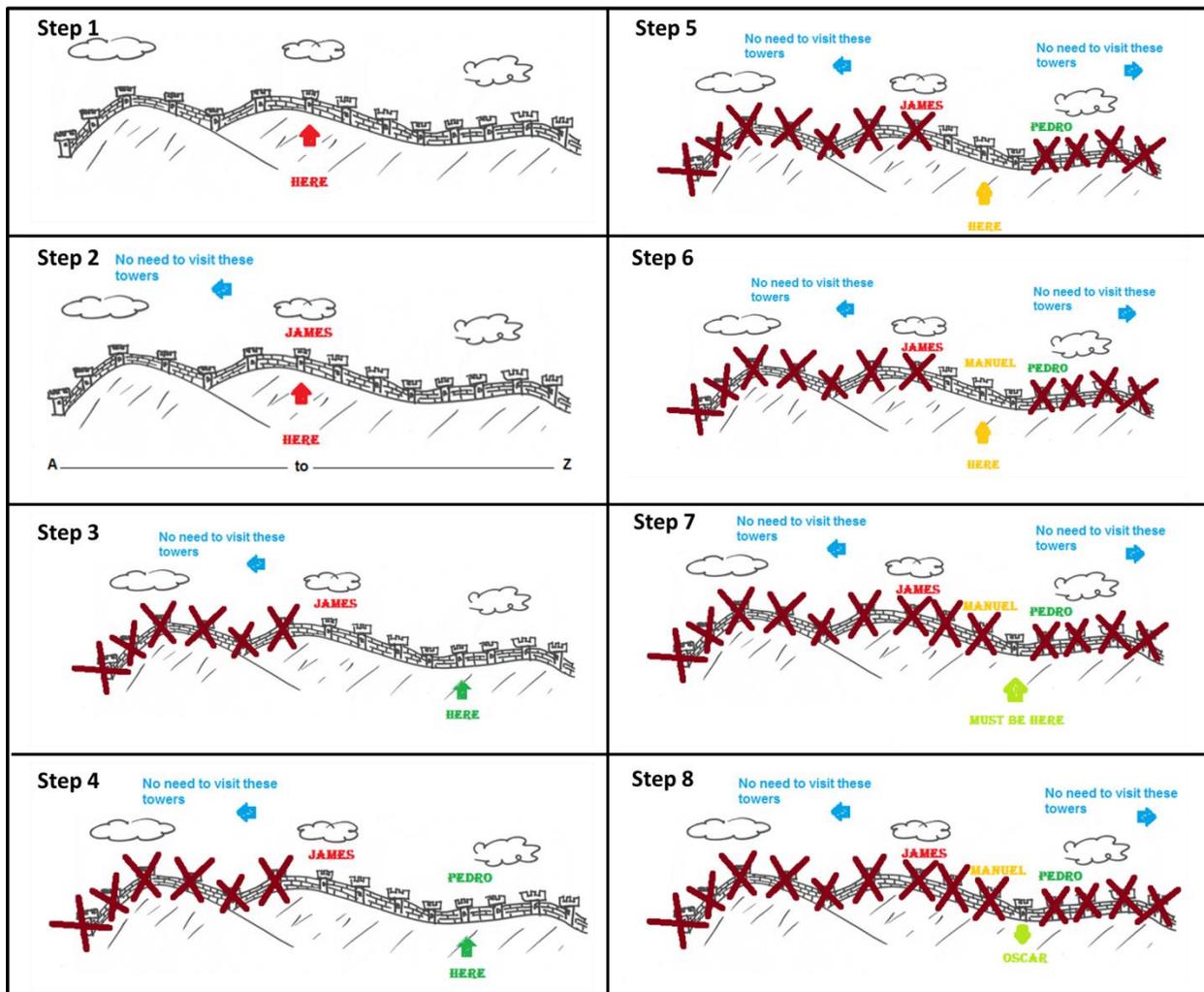


Figure 2.15 Holy Grail Solution

### **Concept of Sorting: Line up according to the month of your birthday (*Activity based Approach*)**

This activity allows students, through silent communication, to sort the participants in the class according to their birthday month. They not only are able to discover new methods of communication but also problem solving skills. The instructions are given to the students as follows: “Without talking, and without asking questions, in 30 seconds, line up according to the month of your birthday. Ready? Go!” Students have to collaborate amongst themselves in order to find a solution. After lining up, the students, in order, mention the month of their birthday and discuss how they were able to find the solution. Most of the time, students perform well and are able to complete the line in the time allotted. A variation of this exercise can be used to introduce students to parallel “merge” sorting.

This activity is followed by a class discussion.

### 3. Assessment

The methods described in this paper were implemented over the course of a semester for a class “Design and Analysis of Computer Algorithms”. To gauge the effectiveness, a questionnaire was constructed to gather student feedback. Overall, the feedback was positive. The students were very receptive to being introduced to concepts in a more intuitive way before learning the material in depth. The students who were surveyed were all enrolled in the Department of Computer Science and Computer Engineering from the College of Engineering in a class titled “Design and Analysis of Algorithms”. As this was the first group exposed to the methodology, the feedback is limited to this survey. Currently, assessments are being made to gauge the reaction of students in various stages of learning Computer Algorithms, preferably assessing them at the beginning of the class and throughout different semesters. This procedure will allow us to analyze what will make the largest impact in improving the learning process. The detailed results of the survey can be found in the appendix.

A five point rating system was used when analyzing the survey feedback data from the students. The responses or answer weights were assigned the following way:

- Strongly Agree (4)
- Agree (3)
- Neutral (2)
- Disagree (1)
- Strongly Disagree (0)
- NA Doesn't Apply

The qualitative analysis of the data received from the survey shows that when students were prompted, if they thought developing intuition was important in order to learn computer algorithms, the average response was 3.26. The class also supported the idea of understanding the relevance of an algorithm before the start of programming as the average score was 3.25. The importance of visualization for learning algorithms scored an average of 3.32. When students were asked if they prefer to be involved in communication-based experiences, hands-on activities, 3D puzzles, and team-based activities to intuitively understand algorithms, the average response was 2.6 and above. The responses to these questions highlight that students were greatly receptive to the goals targeted in the experiment.

It is noteworthy that 83 percent of the class identified themselves as having strong backgrounds in mathematics. This shows that although they were proficient in mathematics, they still felt they benefited from initially learning concepts from an intuitive, math-less approach.

In addition to the questionnaire, the students were asked to provide feedback for future computer algorithms classes:

A1: “As stated above, some more brain teasing activities using some of the algorithms would be fun. The curriculum we have is great already, but having supplemental non-mandatory activities that engages students to work together can be an extra fun way to learn. I've always enjoyed the labs we've had for our classes, and in a way these activities would be similar in style.”

A2: “Possibly try to do a hybrid of the techniques discussed in this lecture and a more traditional lecture-note approach to learning algorithms.”

A3: “Personally, a combination of alternative methods and traditional methods of teaching algorithms would be most helpful. Using the alternative methods such as showing simple examples and discussing the relevance of an algorithm would be a good way to introduce it, followed by learning the pseudocode, and then maybe wrapped up with some examples following the pseudo-code to really drive home the concept, as well as help me to figure out how to implement it in actual code on my own.”

A4: “Relate every algorithm learned to a real world example. Use a visual tool (such as those mentioned in the survey i.e, 3-D puzzles, a visual computer animation, or a hands-on activity) to show how the overall algorithm works, but also at each part of the code/pseudo-code to understand what is happening on each line.”

#### **4. Conclusion and Ongoing Work**

As technology progresses, the way students perceive and learn information changes as a result. To adapt to this trend, methods must be further developed and implemented to ensure a higher rate of success not only in letter grades but students’ understanding and retention of the theories being taught. By catering to popular styles of learning while attempting to keep students engaged, there stands a promising outlook for positive change in the retention of information. By fostering a non-intimidating atmosphere and conveying information in a way that requires little to no pre-requisite, there is a possibility not only for aiding existing students in computer engineering but also for the encouragement for a general audience to be introduced to concepts that otherwise seemed exclusive.

The work is in the preliminary stages and there is much analysis to be conducted as well as further development of the methods employed. However, the feedback so far is positive. The project is part of a larger scope at \_\_\_\_ University to teach other key concepts in the same vein. Future work includes corresponding interactive web pages where students can view graphics and videos including links to relevant information.

## 5. Acknowledgements

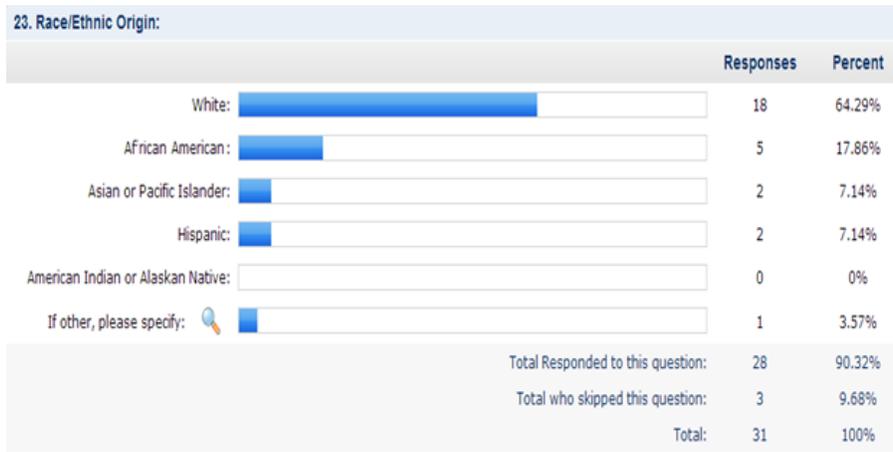
We wish to thank NCIIA and Last Best Chance, LLC for their partial sponsorship of the project.

## 6. References

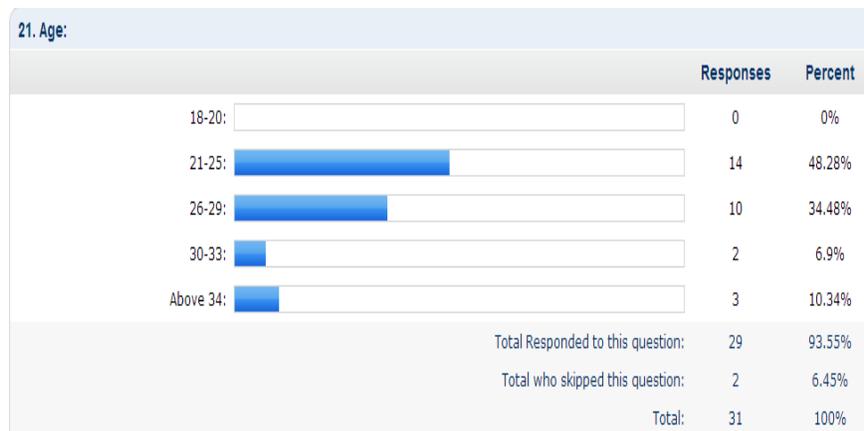
- [1] Mind Tools Limited. (n.d.). “How Your Learning Style Affects Your Use of Mnemonics”, <http://mindtools.com/mneme1stylo.htm>
- [2] Foster, Christopher (2008). *Learning for understanding: Engaging and Interactive Knowledge Visualization*. <https://www.dur.ac.uk/resources/tel/TR-TEL-08-01.pdf>
- [3] Gangwer, Timothy (2009) *Visual Impact, Visual Teaching: Using Images to Strengthen Learning 2<sup>nd</sup> edition*, Corwin Press, CA
- [4] LaFosse, Michael (2009), *Money Origami Kit*, Tuttle Publishing

## Appendix

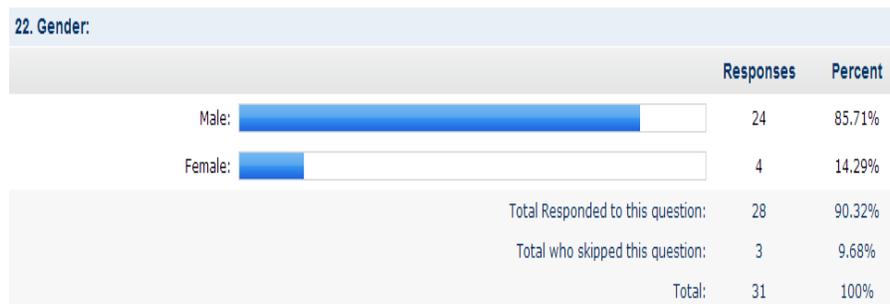
### Demographics



**Figure A.1 Race Distribution**



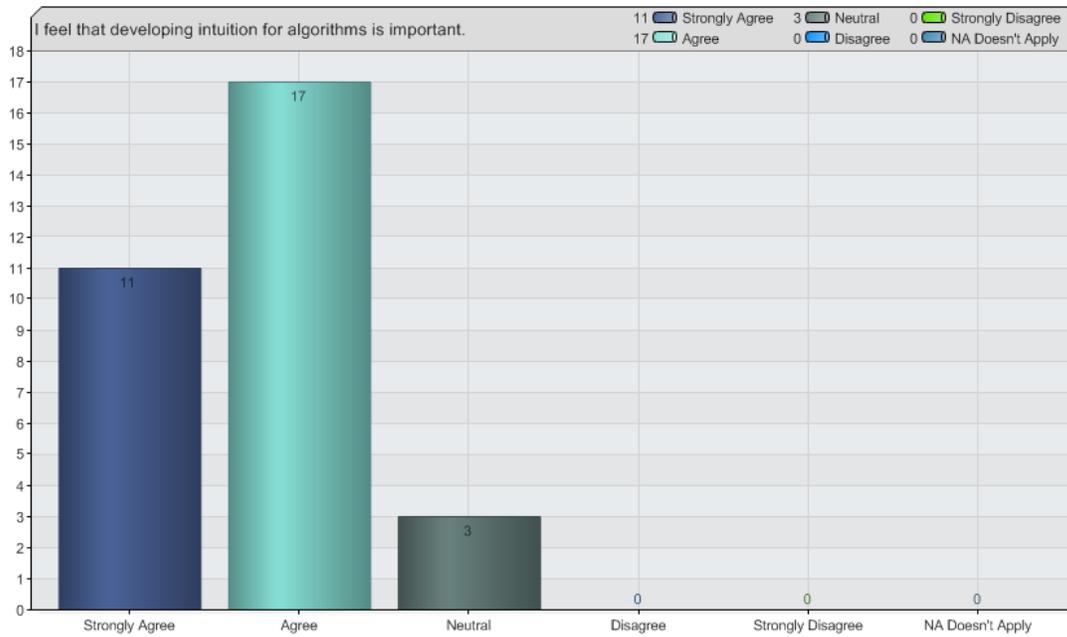
**Figure A.2 Age Distribution**



**Figure A.3 Gender Distribution**

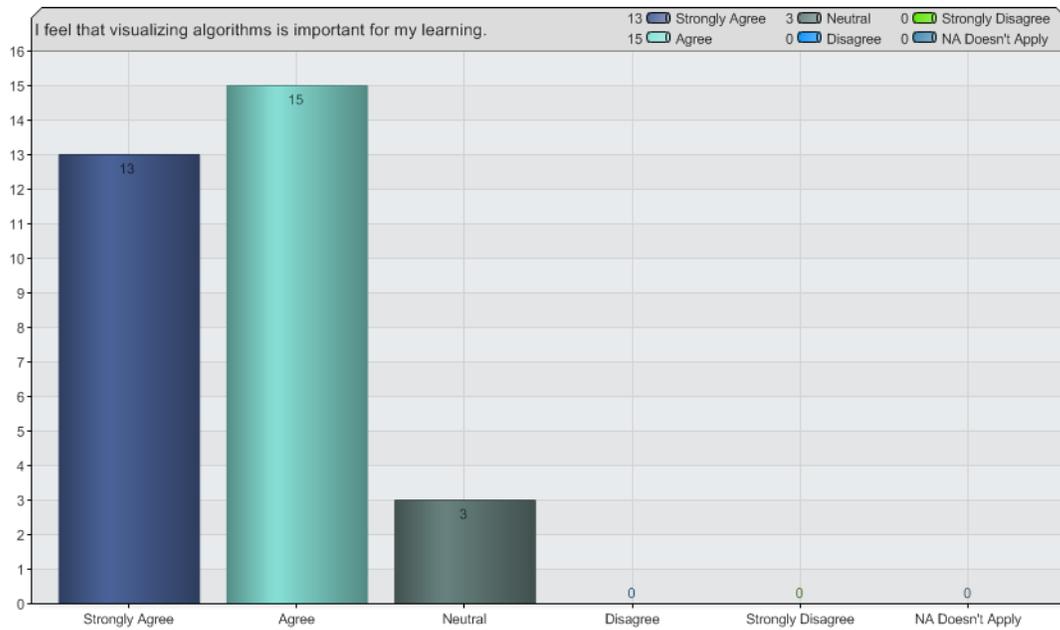
## Questionnaire

Q: I feel developing intuition is important



**Figure A.4** Student Feedback on developing intuition

Q: I feel visualizing algorithms is important for my learning



**Figure A.5** Student Feedback on Visualization of Computer Algorithms

Q: I prefer to be introduced to a new algorithm via hands-on activities

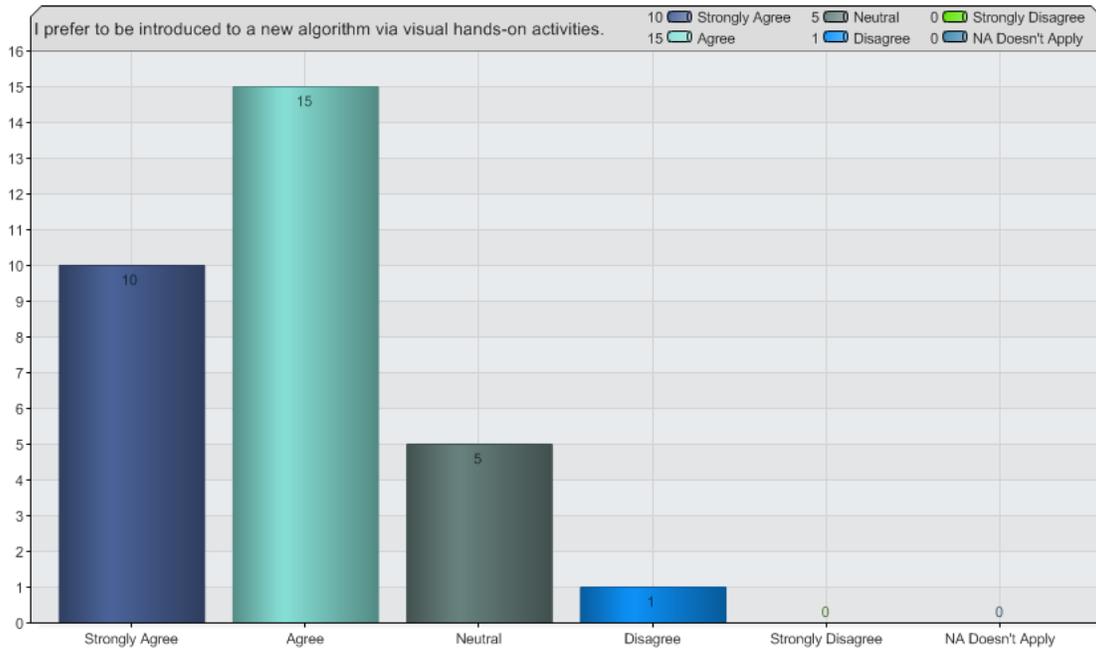


Figure A.6 Student Feedback on hands-on activities

Q: I prefer to be engaged in team-based activities to learn new algorithms

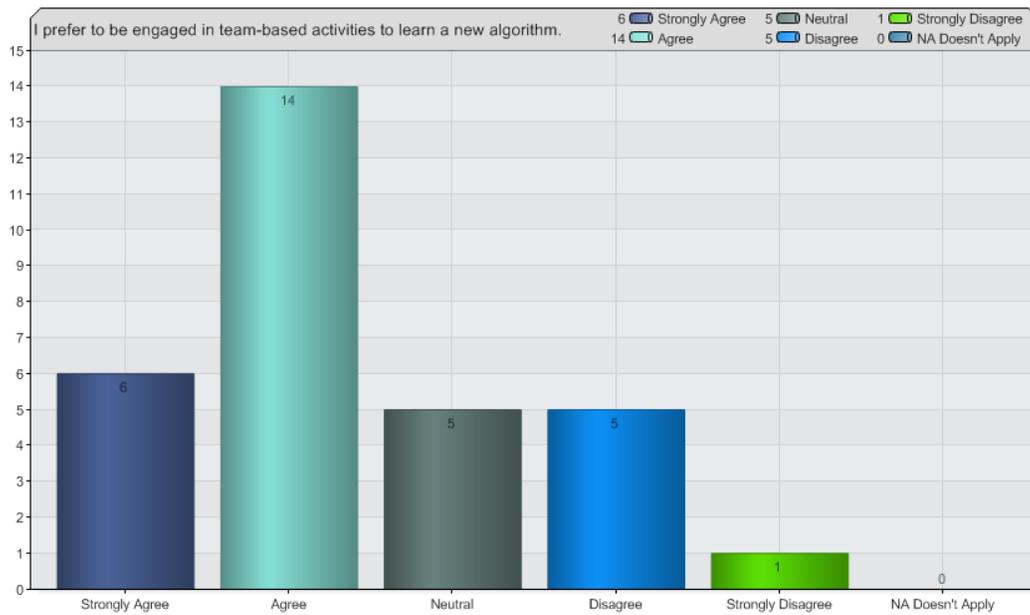


Figure A.7 Student Feedback on team-based activities

Q: I prefer to be involved in communication-based exercises to intuitively understand computer algorithms

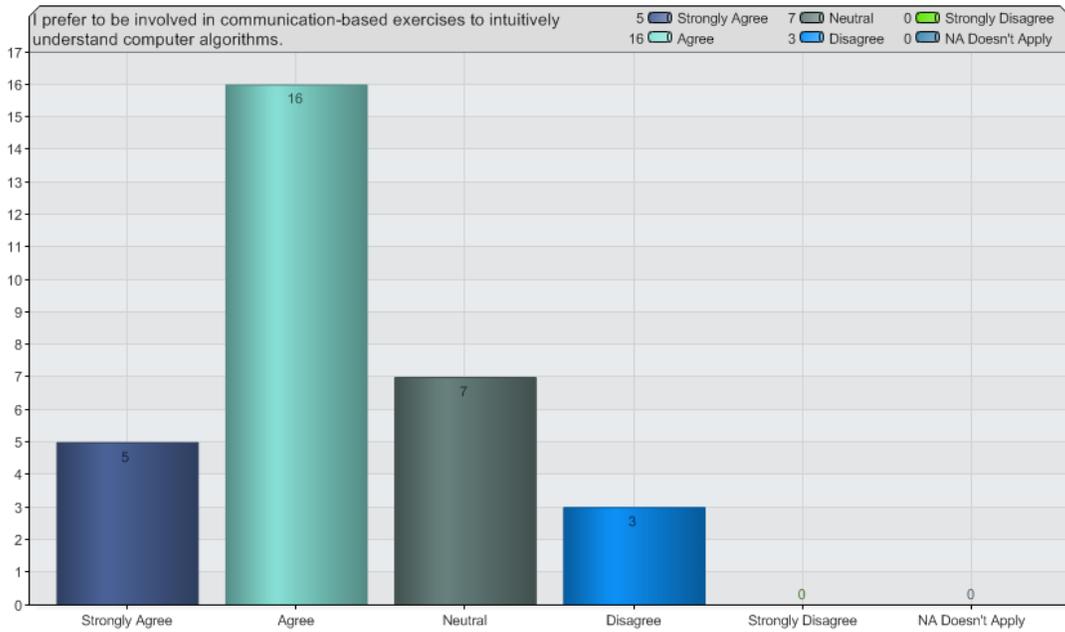


Figure A.8 Student Feedback on communication-based exercises to develop intuition

Q: I prefer to be introduced to algorithms via challenging brain teasers

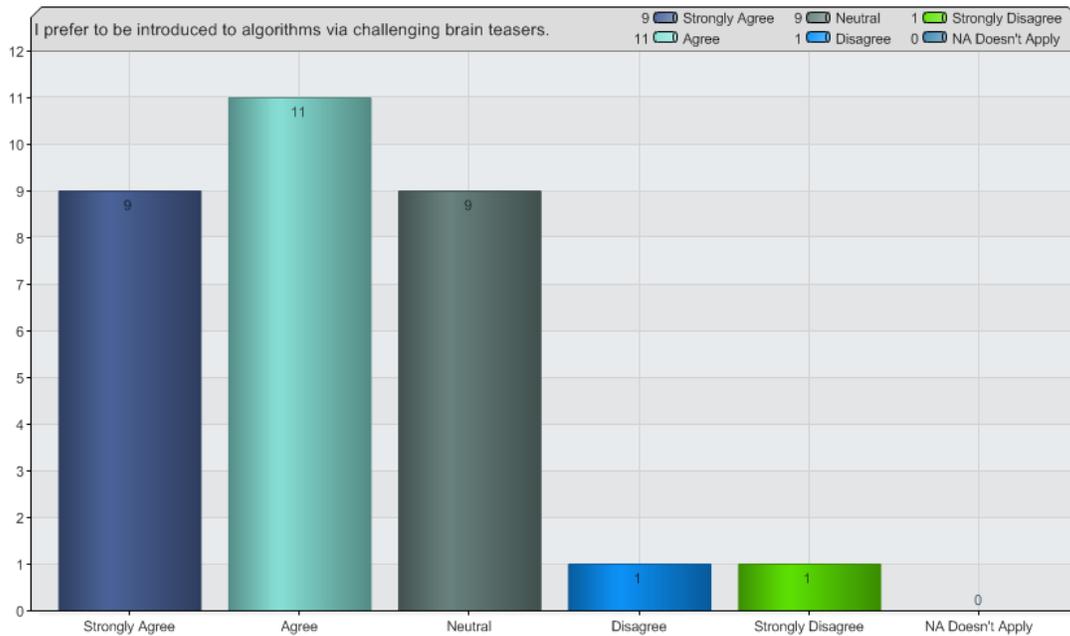


Figure A.9 Student Feedback on brain teasers

Q: I prefer to be taught/learn algorithms using flowcharts

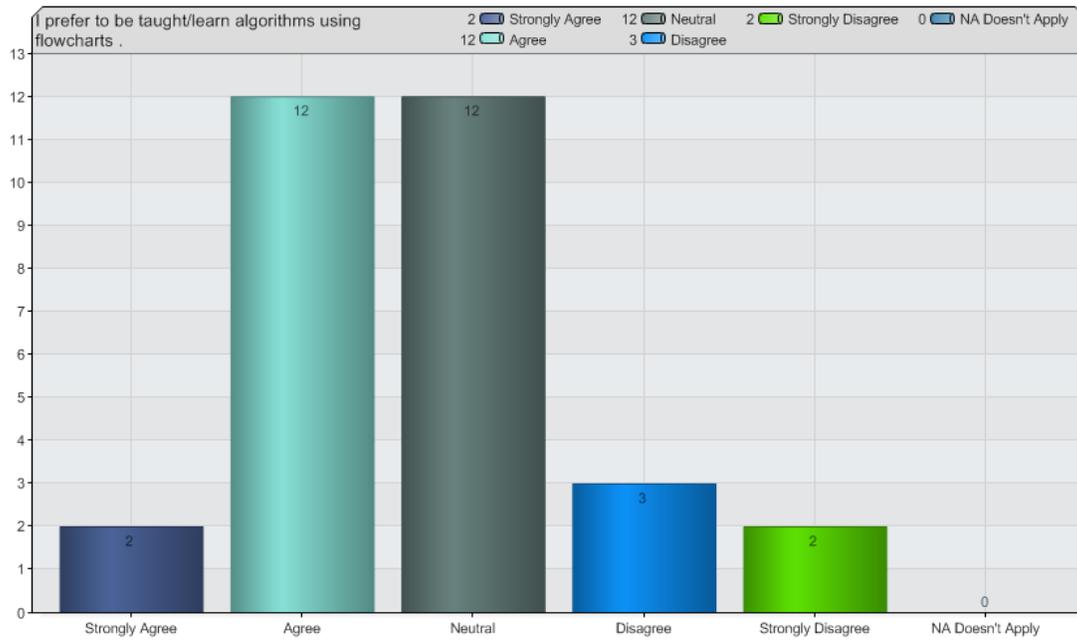


Figure A.10 Student Feedback on using flowcharts to learn

Q: I prefer to taught/learn algorithms by using pseudo codes

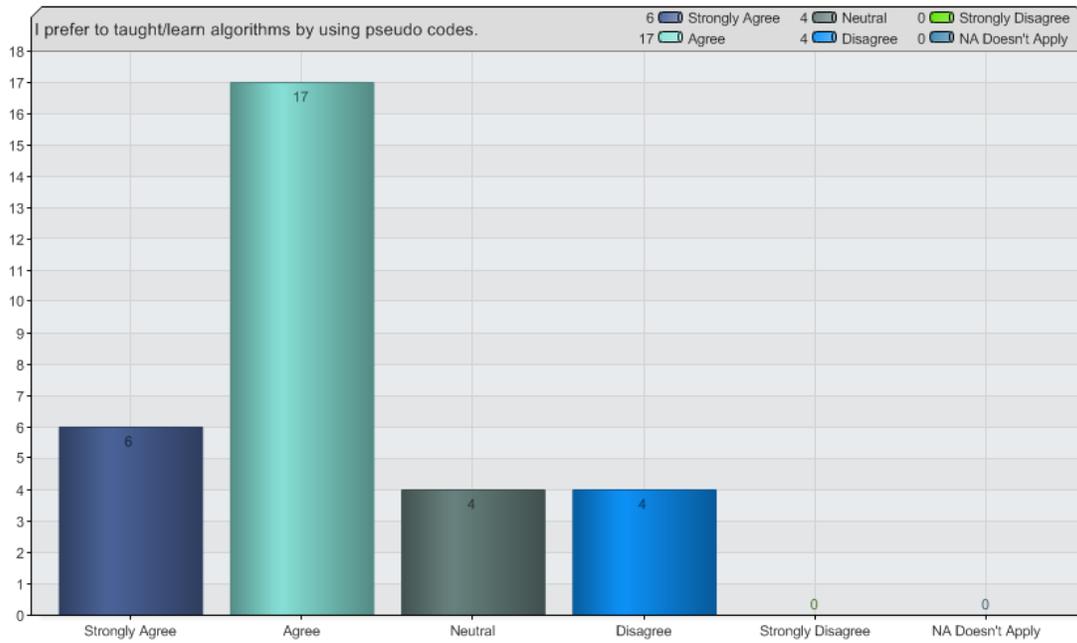


Figure A.11 Student Feedback on using pseudo-codes to learn

Q: I prefer not to be taught. I prefer to learn myself

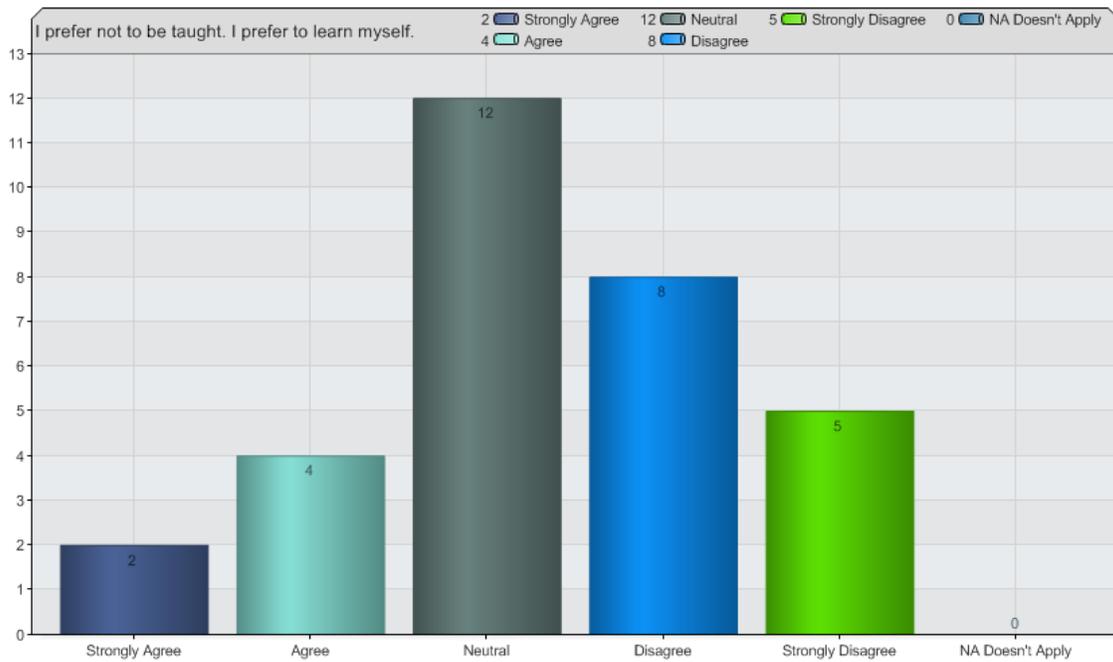


Figure A.12 Student Feedback on learning

Q: I prefer to learn in a fun way

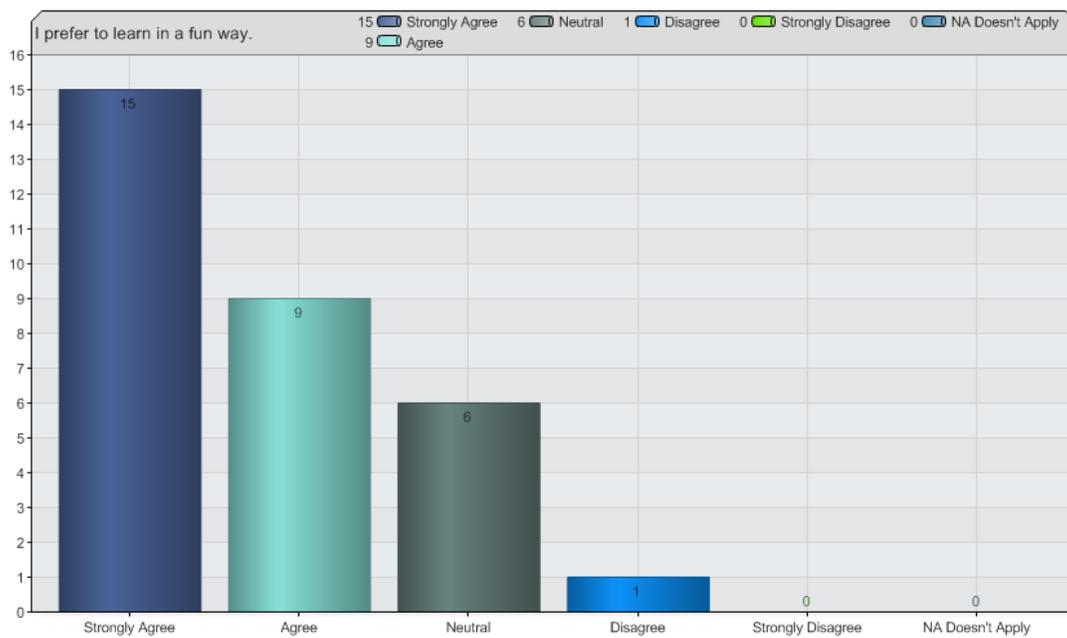
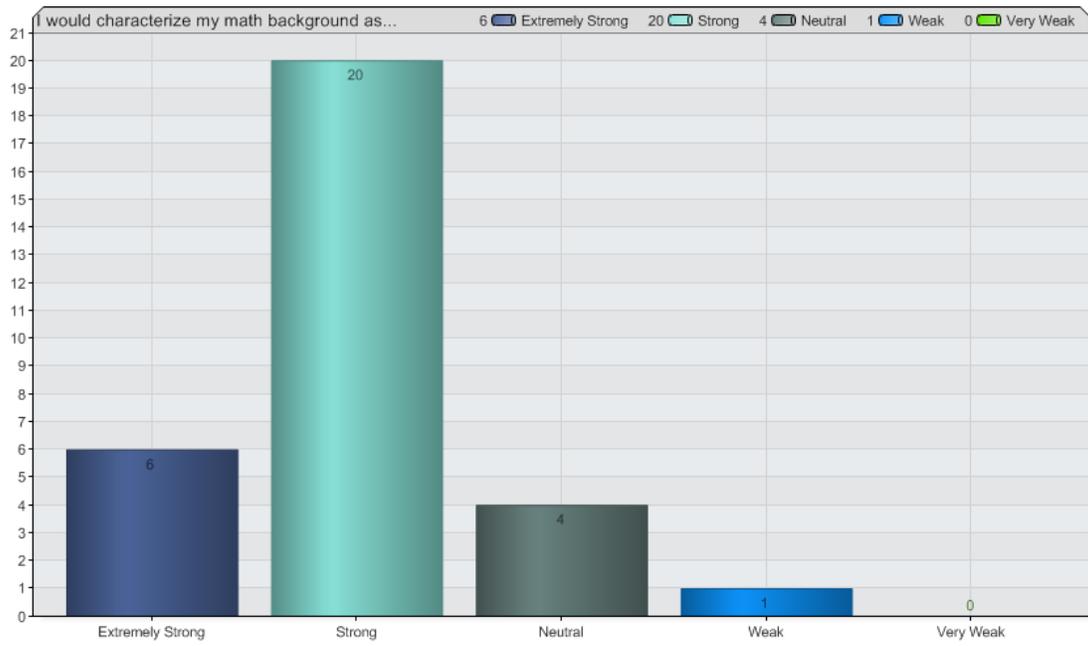


Figure A.13 Student Feedback on learning in a fun way

Q: I would characterize my math background as...



**Figure A.14** Student Feedback on math background