THINKING OUT OF THE BOX: EXPERIENCES FROM AN AFFORDABLE, HOLISTIC AND EMPOWERING COMPUTING COURSE FOR MIDDLE SCHOOL STUDENTS

Elavarasi MANOGARAN[1] , Shoaib DAR[2] , Aashi AGARWAL[3] , Nikhil AGRAWAL[4]

1. INTRODUCTION & NEED:

Computing education in India has been taking a back seat at the K-12 level owing to several reasons such as unavailability of low cost computing devices and lack of access to computer labs and computer science offerings especially in rural settings. Besides the cost barriers, several foundational courses teaching computer science report high cognitive load on novice students especially while learning topics like programming and problem solving that involves complex text-based coding and interpreting syntax. While several introductory computing courses have moved towards a block-based programming approach, only few of these programming environments are able to deliver the holistic objectives of the subject involving hardware and software to interact with the real world in real-time and even if they do, they come with a higher cost, leaving them in-accessible for the under-privileged yet deserving communities. To address this challenge, Pi Jam Foundation, working with its team of industry professionals, curriculum authors, teacher trainers, instructors and student leaders developed an equitable computing tool, **T**hink-**O**ut-of-**T**he-**B**ox (TOTB) to reduce the cognitive overload and entry barriers while fostering problem solving, creative and computational thinking abilities for students in under-privileged and marginalised communities.

2. AN AFFORDABLE & HOLISTIC COMPUTING TOOL

The *think-out-of-the-box* tool aims to reduce access and entry barriers to computing by offering the following components,

Mobile-Based Modular Electronics Kit: a DIY mobile-based modular electronics toolkit contained in a box that is similar in aesthetics to a geometry box, comes with a wide variety of sensors and actuators and an embedded micro-controller board, that enables kids (3 to 4 at a time) to learn programming and tech-enabled problem solving collaboratively at a radically lower cost (cost in vicinity 10-12 dollar per kit excluding the cost of mobile device). Figure 1 shows the kit containing an arduino uno, connecting wires, breadboard and a variety of sensors and actuators. Figure 2 shows the Arduino uno microcontroller connected to a Mobile phone through an OTG cable and an USB port

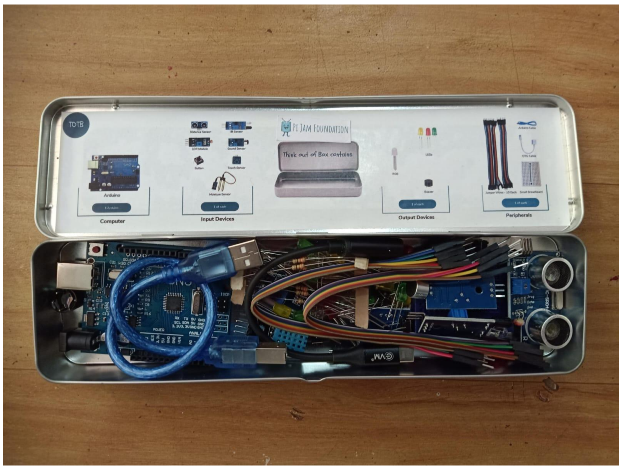


Figure 1. Think out of the box kit containing an arduino uno, connecting wires, breadboard and a variety of sensors and actuators

Figure 2. Arduino uno microcontroller connected to a Mobile phone through an OTG cable and an USB port

Block-Based Programming App with tangible and scannable Paper-blocks: a block-based programming environment that works in conjunction with tangible paper blocks (that resemble jigsaw puzzles) that could be scanned into a mobile device to create code blocks. Paper blocks allow for tangible collaborative play and have the potential to motivate even novice learners to learn computing. Students make use of these paper blocks to create a plan or an algorithm for a project before turning this into code blocks in their app. The code blocks provide instruction to the arduino uno microcontroller. There are tutorial videos embedded within the app to guide students with possible problem statements, solutions, circuit connections and programming. Steps to connect the arduino and phone and to create a new project can be found here. Figure 3 shows a sample paper block code and Figure 4 and 5 shows thumbnails of the app and the programming platform.

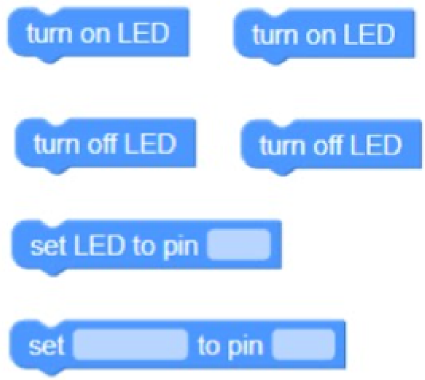


Figure 3. Examples of paper blocks to be used to write a plan or an algorithm

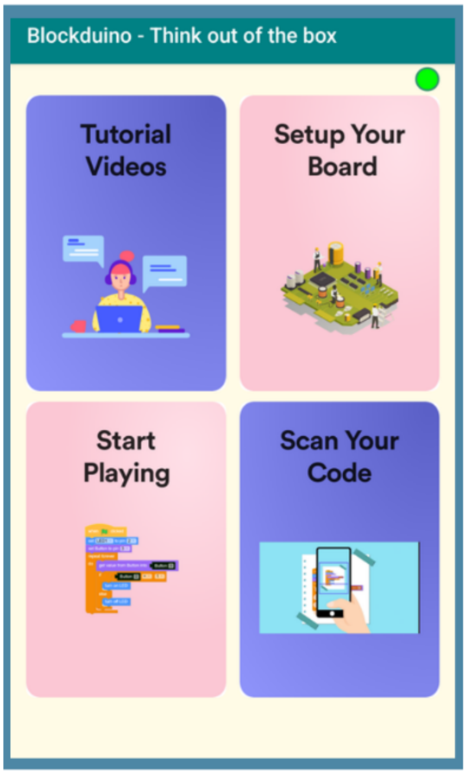


Figure 4. Think out of the box Blockduino app showing tabs for tutorial videos, board setup, creating projects and scanning paper blocks

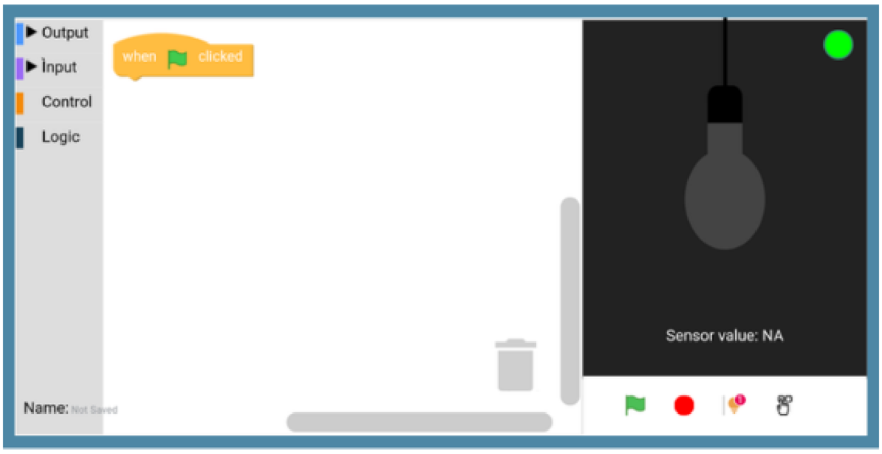


Figure 5. Block-based programming environment of the Think out of the box Blockduino app

The TOTB app currently supports micro-controller boards Arduino Uno, Nano and Raspberry-Pi Pico RP2040

3. PEDAGOGY & ASSESSMENTS

The team realised that for a tool to be truly accessible, one needs to go beyond the dimensions of cost and ease of use, to explore a pedagogy of learning that encourages choices to solutions, choices for representation and expression and to effectively drive engagement of all learners to solve meaningful problems around them. This led to the development of a baseline assessment which studied students' level of accessibility to technology tools, their prior experience with physical computing and problem solving, their creative and computational thinking abilities and their ability to move beyond their limiting beliefs to use technology to solve problems that are critical to their community.

3.1 LEARNING OBJECTIVES:

The learning objectives that were measured during the baseline assessments were as follows,

- ***Accessibility:*** *Students would be able to evaluate their own access and frequency of access to technology tools and resources, evaluate their prior knowledge of solving problems using technology and also identify the choice of tool that best suits their needs of and learning*

- ***Computational Thinking:*** *Students would be able to create programming projects* *that involve computational thinking concepts like sequencing, loops, selection etc, apply skills like algorithmic thinking and programming, demonstrate practices like debugging and remixing and develop attitudes like collaboration, confidence, learning from mistakes etc while solving real-life problems*

- ***Empowerment:*** *Students would be able to analyse and justify their limiting beliefs and situations about technology and computer science, use design thinking principles to explore their community's critical needs and think about taking the agency to solve those needs using technology. Students use technology as an emancipation and humanization tool to solve problems that are most pressing to their local context*

In addition to this, we also conducted the *Problem Solving & Creativity Framework and Micro-assessment which* acts as a formative self-assessment tool for students to assess their problem solving and creative thinking skills after every class. For the micro-assessment, we started by defining the set of skills and thinking processes that are necessary to foster students' learning and practice of creative thinking and problem solving skills as follows:

- ***Problem Solving*** *is a set of skills and techniques involving understanding the problem, representing and formulating the problem, planning and executing the problem,monitoring the progress and reflecting on feedback to fix challenges.*

- ***Creativity*** *is a way of thinking and practice that involves exploring a variety of choices to problems and solutions, expressing unique ideas and opinions (and) meaningfully engaging in ways that bring more joy and a sense of purpose.*

Next in an effort to scaffold these skills and thought processes into classroom instruction, we created a table that encompasses a sequence of objectives for each skill and some reflection questions for classroom talk where teachers use the following questions to foster student learning during every class

Table 1: Objectives for Problem Solving & Creativity and reflection questions for each category

Category	Reflection questions
1. Understanding of the Problem ***Problem Solving***	• What problem are you trying to solve today? Who is benefited by this? • Why is this problem important to your community?
2. Exploration of choices to	• What will you create/solve today?

problems/solutions ***Creativity***	• What are similar problems? Why did you choose this problem? • What are possible solutions? Why did you choose this solution? • What connections did you make (between problems or solutions) to arrive at your solution?
3. Expression **Creativity**	• Could you explain/express your unique solution to others in your favourite way? • Could you explain how this solution is novel or new? • What tools or things would you use to solve this problem?
4. Representation **Problem Solving**	• What tools/objects would you need to solve this problem? • How would you use these tools to solve them?
5. Planning and Execution **Problem Solving**	• What does your plan and program look like? • What role do you play to solve the problem?
6. Reflection **Problem Solving**	• What's your favourite part about solving the problem? • How did you fix challenges? • Which team/friend did you help? • What did you like about the other team's work? Do you have any ideas for them to improve?
7. Engagement **Creativity**	• What is one thing, person, situation or thought that made you happy/joyful and immersed in class today? What made you go with the flow? • Why did it make you happy?

Teachers use the objectives and questions from Table 1 to check students' progress in each class in terms of the strategies they use to solve problems (creative process) besides the product (solutions) in that specific order.

Students write their responses in their journal or notebook and maintain a record of the problem they explored, the plan and their creative process in each class.
We also recommend teachers to maintain a record of students who are doing well with problem solving and those who need further support. The same goes with students who expressed themselves creatively and those who need additional support. Teachers also keep track of students who expressed interest in solving critical problems in their community and students who exhibited exceptional computational thinking skills like algorithmic thinking, while also encouraging those who lack attitudes like collaboration or

perseverance. A copy of the record can be printed from here and used by the teacher during every class. A sample of this record with examples can be found here.

At the end of every session, after reflecting on questions from Table 1, students take a self-assessment which evaluates them against the objectives as mentioned in the table. The assessment requires students to choose between different learning levels based on the problem solving strategies they used and their creative process, including skills like collaboration, critical reflection and engagement. The self-assessment rubric which includes the level of attainment (Beginning, Developing, Exemplary) and the grading criteria could be found here.

The Think Out of The Box course is rapidly spreading in response to demand from schools nationwide, from industry leaders and professional organisations. We have currently reached 290+ students across our nation, in Maharashtra and Kashmir, including a pilot in Kenya. Figure 6 shows a TOTB classroom in India (left) and Kenya (right).

Figure 6. Think out of the box pilots in India (Left) and Kenya (Right)

3.2 BASELINE ANALYSIS

We conducted a baseline assessment to evaluate how our students perform across our objectives and to identify where they need help in terms of accessibility and solving problems that are relevant to their community.

Some of the observations and inferences we made could be found below,

ACCESSIBILITY:

Out of 165 middle school students who took the baseline assessment in India, 71.5% of the students feel problem solving with technology is very important for their communities. Yet as shown in Figure 7, ***the proportion of students who reported using technology for problem solving is relatively lower on a daily, weekly or monthly basis.*** This indicated that regular access to affordable technology tools is challenging for these students' contexts.

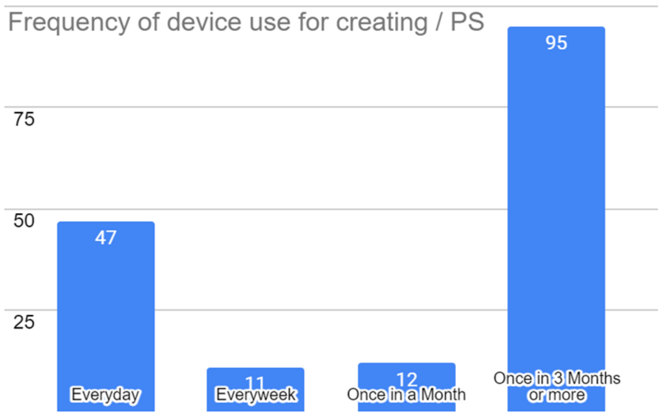
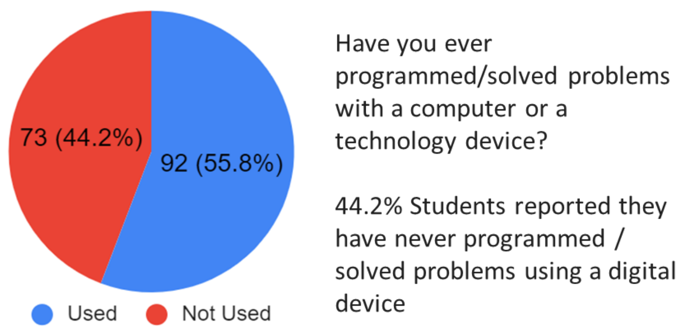


Figure 7. Number of students creating or solving problems using technology everyday, every week, once in a month and once in 3 months or more

Holistic approach to Problem Solving using Technology:

As a part of measuring accessibility, our baseline assessment was also aimed at understanding if students already used a computer or a technology device to solve problems or program a technology device. We also asked students if they programmed hardware devices such as micro-controllers or sensors.



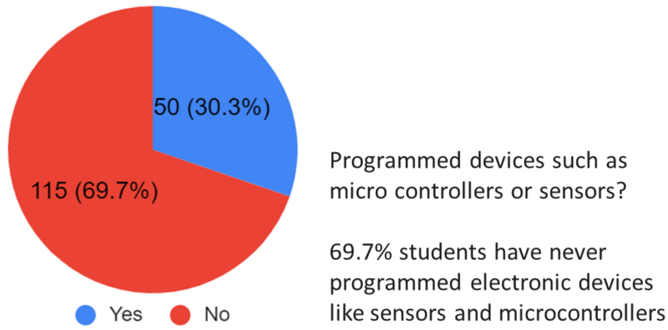
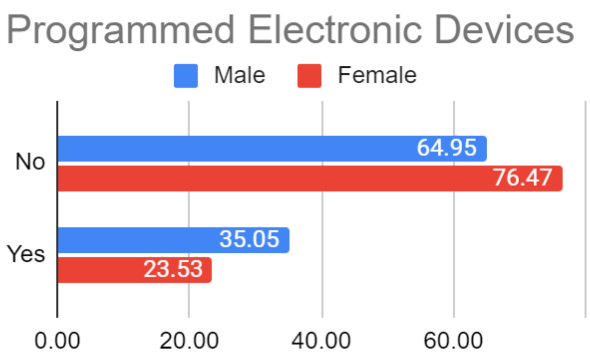


Figure 8. (Top) Percentage of students who have used/not used a computer to solve problems. (Bottom) Percentage of students who have answered yes/no to whether they have programmed devices such as micro-controllers or sensors



Programmed Electronic Devices	Male	Female
No	64.95	76.47
Yes	35.05	23.53

Figure 9. Percentage of male and female students who have answered yes/no to whether they have programmed devices such as micro-controllers or sensors

As shown in Figure 8, although 55.8% of students have reported using a technology device to solve problems, ***69.7% of total students said they have never programmed electronic devices like sensors and microcontrollers***, showing that the majority of students lacked a holistic perspective of using both the hardware and software tools to solve real-life problems. As shown in Figure 9, ***a significant proportion of females (76.47%) lack experience in working with electronic devices calling for attention to equitable access to technology and physical computing devices.***

Also owing to a greater percentage of students who have reported solving problems with technology in their past, we aim to clarify with students what their definition of problem solving is and get more clarity on the problems they solved, keeping in mind the prevalent confusion that exists among novice learners in this context, on assuming 'consumption of technology' as 'solving problems using technology'.

As we believed that accessibility is strongly linked to giving students the choice of tools that they are inclined to use, we asked students for their tool preferences for each of the learning scenarios and as shown in table 2. ***For all learning scenarios, except debugging, code blocks were the first tool choice, with female students indicating a stronger preference. Paper blocks were the second most preferred tool, with male students indicating a stronger preference. Male students in fact preferred paper blocks as the most preferred tool for debugging, i.e finding mistakes in their code.***

Table 2: Tool preferences of students when given a choice of using code blocks, paper blocks or electronics for a given learning objective

Learning scenarios	**Male Learners (Preference 1)**	**Male Learners (Preference 2)**	**Female Learners (Preference 1)**	**Female Learners (Preference 2)**
Problem Solving	Code Blocks (45.8%)	Paper Blocks (36.5%)	Code Blocks (61.2%)	Paper Blocks (31.3%)
Programming	Code Blocks (54.3%)	Paper Blocks (24.5%)	Code Blocks (71.6%)	Paper Blocks (17.9%)
Algorithmic Thinking	Code Blocks (50.5%)	Paper Blocks (24.7%)	Code Blocks (63.2%)	Paper Blocks (19.1%)
Debugging	Paper Blocks (44.6%)	Code Blocks (32.3%)	Code Blocks (47.6%)	Paper Blocks (36.5%)

In general, electronics were the least preferred tool across all four learning scenarios and hence this is not mentioned explicitly in the table above. The reason for this could be attributed to the majority of students not having any prior experience working with electronics. Noticing that electronics were students' least choice, we wanted to cross verify if this is because of their lower levels of confidence and we *observed something interesting.* ***Please see the following section on 'Limiting Beliefs' which talks about students' confidence levels in working with electronics.***

What we inferred:

ACCESSIBILITY (Baseline):

- *71.5% of the students feel problem solving with technology is very important for their communities, yet the proportion of students who reported using technology for problem solving is relatively lower on a daily, weekly or monthly basis*

- *69.7% of total students said they have never programmed electronic devices like sensors and microcontrollers with a significant proportion of females (76.47%) lacking experience in working with electronic devices, calling for attention to equitable access to technology and physical computing devices*
- *Out of code blocks, paper blocks and electronics, code blocks were the most preferred tool choice for learning skills like programming, problem solving and algorithmic thinking. Paper blocks were the second most preferred tool except for debugging where it was reported as the first choice. Electronics was the least preferred choice which could be attributed to the lack of prior exposure reported above*

EMPOWERMENT:

We wanted to study students' limiting beliefs and situations about technology and computer science. In addition to this, we also wanted to study if our students are aware of their communities' needs and if they think about taking the agency to fulfil those needs using technology. Students were asked to record their responses to 10 questions that evaluated their ability to become empowered learners. These questions asked students how confident they feel about solving problems with technology besides asking them to identify any problem in their immediate community and think about ways to solve them using technology.

Limiting beliefs:

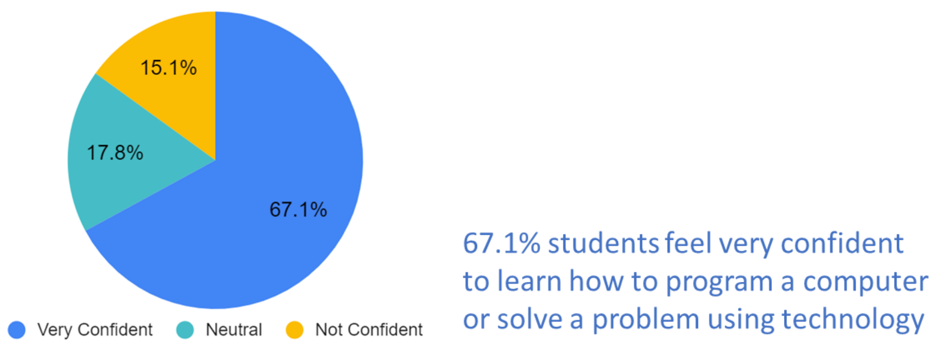


Figure 10. Percentage of students who are confident to learn how to program a computer or solve a problem using technology

Despite the majority of students having less exposure to physical computing, 67.1% of students felt very confident to learn how to program a computer or solve a problem using technology. ***This shows that most students already are intrinsically motivated about***

solving problems using technology. Females tend to express slightly more confidence levels in working with electronics with 61.8% of female students expressing their confidence vs 56% males. This is irrespective of the fact that the majority of female students did not have any previous exposure to electronics. Also, students tend to prefer electronics as the least choice not because they were less confident with the tool, but because when given a choice of using these tools, (refer table 2) they prefer to get started with code blocks and paper blocks before proceeding with electronics. Hence we infer that enabling equitable access to physical computing devices in conjunction with block programming and tangible paper blocks has the potential to further improve their confidence levels. On the other hand, decline in confidence levels while problem solving with technology could be attributed to the complexity and cognitive load of the computing tools, relevance of curriculum resources, instruction or learning methods over the period of intervention.

Table 3: Number of students who responded to reasons on why does learning computer science frighten them

Why does learning computer science frighten you?	Count
	0
I am not good at coding.	24
I do not know anything about it.	51
I'm afraid of making mistakes and failing.	34
It is difficult and challenging	27
Others	1
Grand Total	**137**

Table 3 shows that out of all students who said that learning computer science frightens them, the most common reasons were " I do not know anything about it" and "I'm afraid of making mistakes".

Identifying problems relevant to their community:

In our analysis, it is evident that both male and female students recognize the importance of technology-driven problem-solving, with 71.5% of the overall student population considering it crucial.

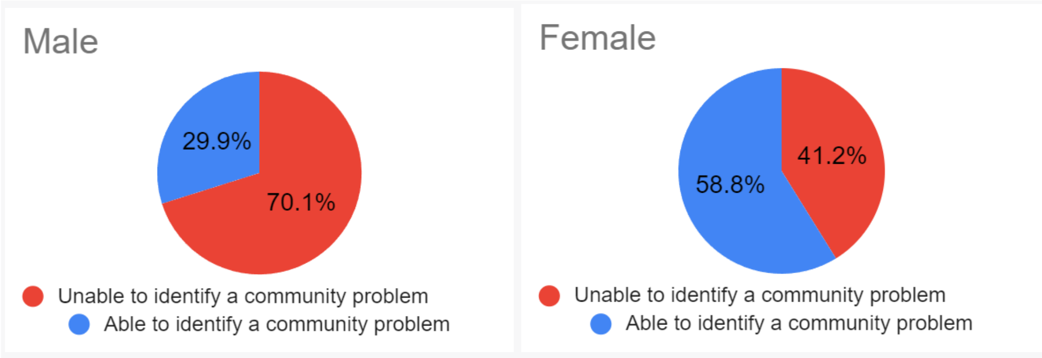


Figure 11. Percentage of male and female students who are able to identify critical problems to solve in their community

Females, however, exhibit remarkable enthusiasm for learning computer science, as 70% express excitement compared to 47% of males. They also excel in community problem identification, with 58.8% of females (as shown in Figure 11) identifying relevant community issues and explaining technology's role in solving them, showcasing their strong empowerment potential. Yet as described in Figure 9, female students lack exposure to the technology tools and learning resources to allow them to fulfil their potential.

What we inferred:

EMPOWERMENT (Baseline):

- *Females tend to express slightly more confidence levels in working with electronics with 61.8% of female students expressing their confidence vs 56% males. This is irrespective of 76.47% of females not having any prior experience learning electronics and could be attributed to their intrinsic motivation.*
- *Although more students express higher confidence levels working with electronics, when asked for the choice of tool to get started with problem solving & programming, electronics was their least preferred tool, with code blocks and paper blocks being most preferred. Hence we infer that enabling equitable access to physical computing devices in conjunction with block programming and tangible paper blocks has the potential to further improve their confidence levels*
- *70% of females express excitement to learn computer science and 58.8% excel in community problem identification. Yet a majority of females, i.e 76.47% report not having programmed any electronic devices in the past, lacking exposure to technology tools and learning resources*

COMPUTATIONAL THINKING LITERACIES:

Out of the 165 students who took the diagnostic pre-test, an average score of 4.23 was reported out of a total of 11 marks. Only 12% of students answered correctly for the question on pattern recognition and 20% of the students gave correct responses for the question on algorithmic thinking. Questions on sequencing and repetition got 40% and 12.7% correct responses respectively. This suggests that students need more support on computational thinking skills and concepts. Only 12% of students were able to respond correctly for a question on coding for a sensor using selection logic, which infers that students lack understanding of core concepts of programming and physical computing.

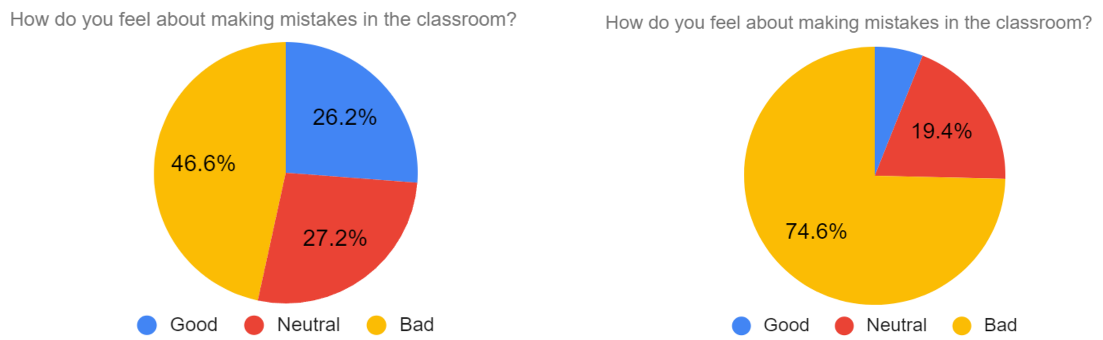


Figure 12. Percentage of male (left) and female (right)students who responded to the question "How do you feel about making mistakes in the classroom?"

As shown in figure 12, ***more female students express feeling bad about making mistakes in the classroom when compared to male students***.

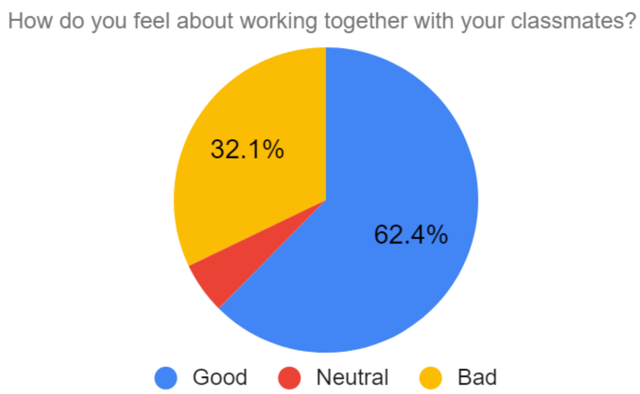


Figure 13. Percentage of student responses for the question, "How do you feel about working together with your classmates?"
32% of students feel bad about working together with their classmates calling attention to teaching students about the power of collaborative problem solving.

About 50% and 60% of students on average reported that they ask for help from others and their classmates seek their help from them respectively, while solving problems. But when asked about the names of their classmates whom they offered help and seeked help from, only as few as 20% to 40% were able to remember names.

What we inferred:

***COMPUTATIONAL THINKING (CT) LITERACIES (Baseline)*:**

- Out of the 165 students who took the diagnostic CT pre-test, an average score of 4.23 was reported out of a total of 11 marks
- Only 12% of students answered correctly for the question on pattern recognition and 20% of the students gave correct responses for the question on algorithmic thinking
- Questions on sequencing and repetition got 40% and 12.7% correct responses respectively
- Only 12% of students were able to respond correctly for a question on coding for a sensor using selection logic, which infers that students lack understanding of core concepts of programming and physical computing
- More female students report feeling bad about making mistakes in the classroom

PROBLEM SOLVING & CREATIVITY:

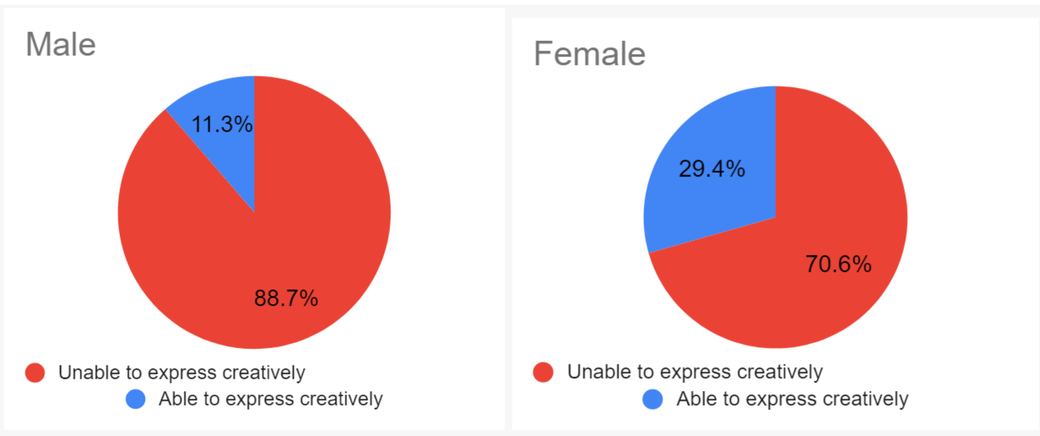


Figure 14. Percentage of male and female students who are able to express creatively when asked the question, "How did you get your name? What's the meaning of your name? Design something creative with your name (You can draw, write a story, poem, song etc)"

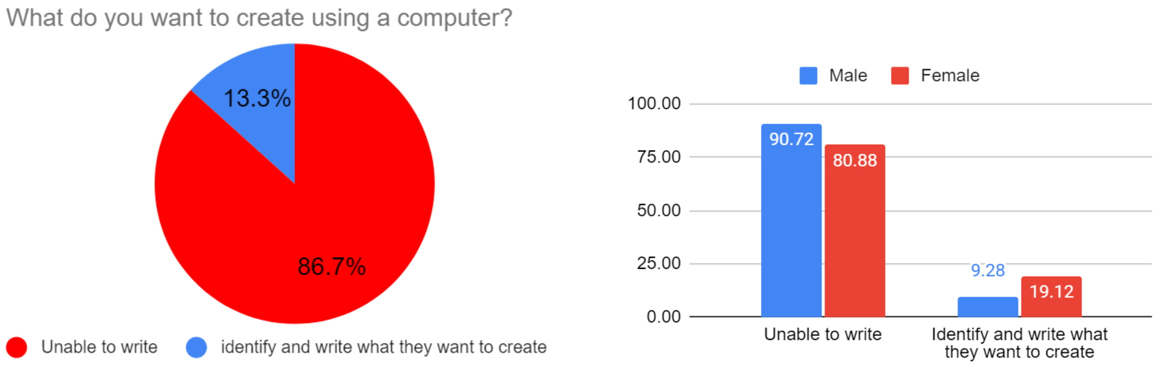


Figure 15. (Left) Percentage of students who were able/unable to respond clearly to the question, “What do you want to create using a computer? Think about your favourite hobby. You can create anything that you like using a computer - a story, drawing, a game etc. Be specific and discuss what your story/drawing/game is. Note: Using gmail, whatsapp, youtube or facebook doesn't count as creating” (Right) Chart shows percentage of male and female students who were able/unable to answer the question

As shown in Figure 14 and 15, female students are more inclined towards creative expression and exhibit stronger creative tendencies while specifying what they wish to create using computers. Overall, providing additional support for male students in community problem identification and ensuring equitable access to electronic devices could further enhance their empowerment. Additionally, from Figure 14, we found that nurturing creativity in both genders through tech-related creative projects that support students' interest areas is important for fostering a well-rounded creative mindset.

The problem solving and creativity self assessment was used as a formative assessment after every class to evaluate problem solving and creative thinking skills of students. All pilots recorded student self-evaluations after every class. An average of the scores of first two classes were taken for four different pilots in Mumbai, Pune, Kashmir and other rural schools and we found that students scored themselves 45% or less in both problem solving and creative thinking objectives. As we see consistency of this data across our pilots, we look forward to capturing and analysing these self-assessments towards the end of the pilot to see how the intervention helped with improvement of these skills. While self-assessments could prove to be an effective formative assessment tool to foster students' learning, it is associated with challenges of students not given due attention to the rubrics or awarding themselves very high or low marks based on their confidence levels. We suggest that teachers should play an integral role in setting the assessment standards clearly in the very beginning and explain to students the objectives and outcomes of the self- assessment, to be able to obtain useful data both to capture learning and enhance their learning.

What we inferred:

PROBLEM SOLVING & CREATIVITY (Baseline):

- *86.7% of students are unable to answer the question 'what do you want to create using a computer?'*
- *More than 80% of the students were unable to express their names creatively when asked to design something creative with their name*
- *Female students are more inclined towards creative expression when compared to males and exhibit stronger creative tendencies while specifying what they wish to create using computers*
- *An average of the scores of first two classes were taken for four different pilots in Mumbai, Pune, Kashmir and other rural schools and we found that students scored themselves 45% or less in both problem solving and creative thinking objectives.*

4. RESULTS OF INTERVENTION:

To identify the impact of our intervention, we conducted endline assessments to evaluate improvements in student learning across the objectives mentioned above. We wanted to study if our intervention is able to deliver improved learning outcomes through specific research questions as described below,

RQ1 (EMPOWERMENT) - Does the think-out-of-the-box approach improve confidence levels in novice learners, especially girls?

We wanted to know if using paper blocks in addition to code blocks would improve confidence levels of novice learners, especially girls, and help them overcome any entry barriers. We asked two questions to see how using paper blocks helped them and found that more female students than males reported paper blocks to be most helpful.

RQ1.1: How helpful was using paper blocks in reducing fear about technology?

# Female	# Male
61	28

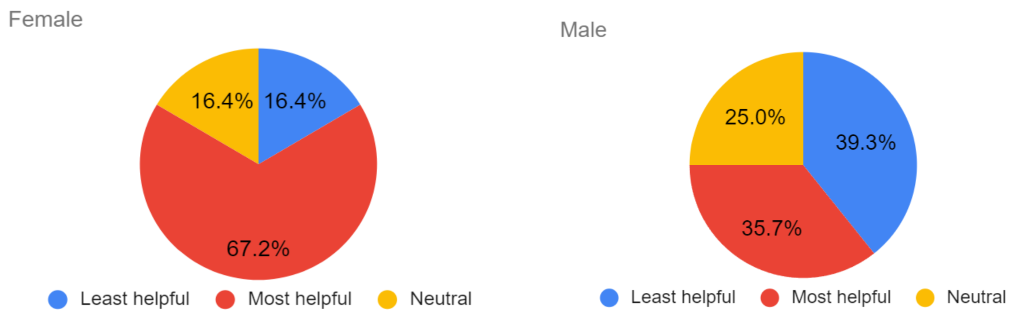


Figure 15. Students' responses to the question "How helpful was using paper blocks in reducing fear about technology"

We asked students who reported "learning computer science frightens them", "How helpful was using paper blocks in reducing their fear about technology" and about ***67.2% of females, out of a total of 61 female students who took the post-test said it was most helpful as shown in Figure 15*** **.** Given only 28 male students took the post-test, we were not able to conclude whether it was perceived useful or not looking at the percentages.

RQ1.2: How confident are you in working with electronics such as light bulbs, wires, sensors and circuits?

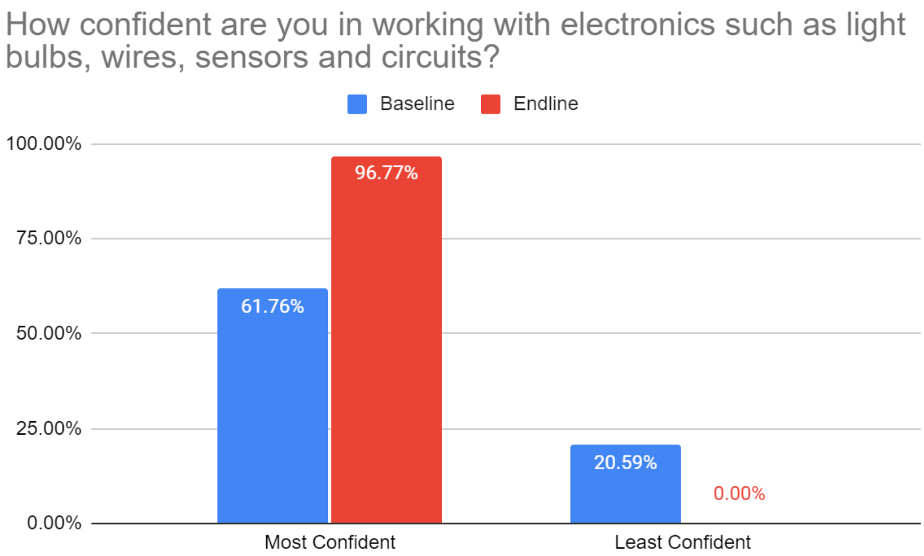


Figure 16. Percentage of female students who responded to the question "How confident are you in working with electronics such as light bulbs, wires, sensors and circuits"

The number of girl students who responded that they are most confident with electronics increased by 35% after the intervention and none of the girls responded they are least confident.

RQ1.3: Does learning Computer Science excite you?

The number of male students who reported that learning computer science excites them improved by 49% and the number of female students who reported the same improved by 23% with less than 10% of students reporting that learning CS frightens them.

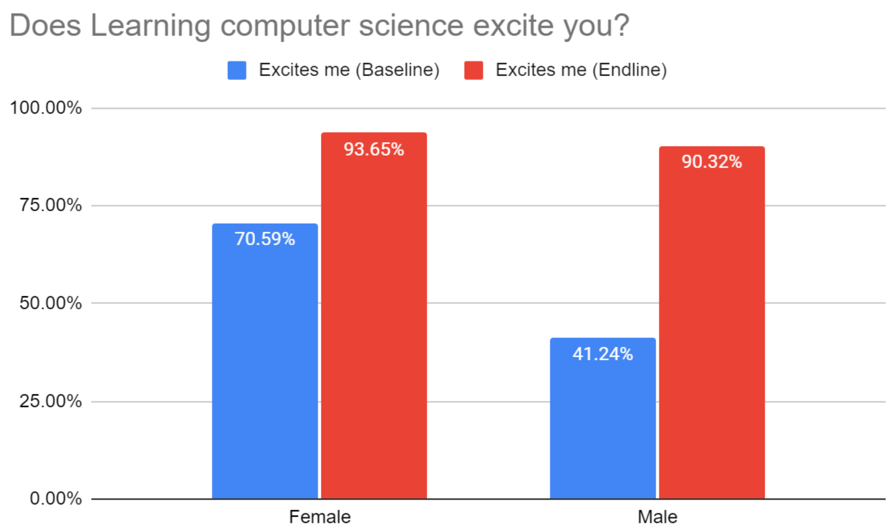


Figure 17. Percentage of female and male students from baseline and endline who said learning computer science excites me

**Check for Aspiration*

RQ2 (ACCESSIBILITY & COGNITIVE LOAD): Does the block-paper-mobile based approach reduce novices cognitive load of problem solving and creating with technology?

RQ2.1: How difficult was programming an arduino uno?

Several foundational courses teaching computer science report high cognitive load on novice students especially while learning topics like programming and problem solving [1][2]. We asked students what they found to be true while programming with their smartphone and ***more than 50% of students found that using a touchscreen is easier than using a keyboard or mouse***.
We also asked students how difficult was programming an arduino uno with a smartphone and the results can be seen in Figure 18.

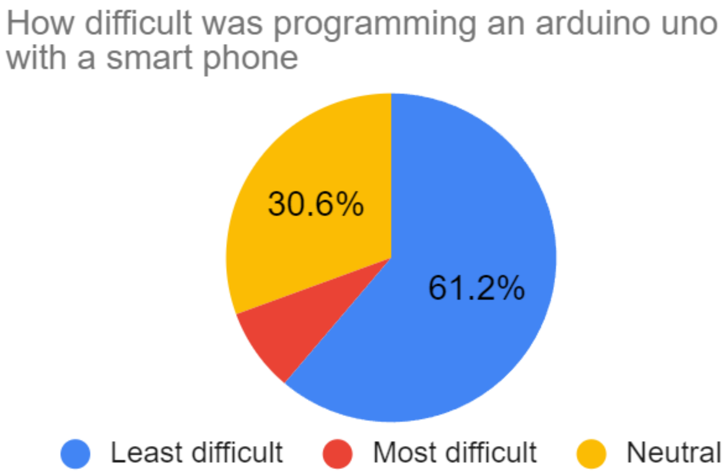


Figure 18. Percentage of students who reported programming an arduino uno with a smartphone as least difficult, most difficult and neutral.
Out of 85 students who took the post test, 61.2% reported that programming an arduino uno with a smart phone was least difficult and only 8% reported that it was most difficult.

RQ2.2: How helpful was using paper blocks in getting started with your project?

We asked students, “How helpful was using paper blocks in getting started with their project and about 67.2% of female students said it was most helpful.

# Female	# Male
61	26

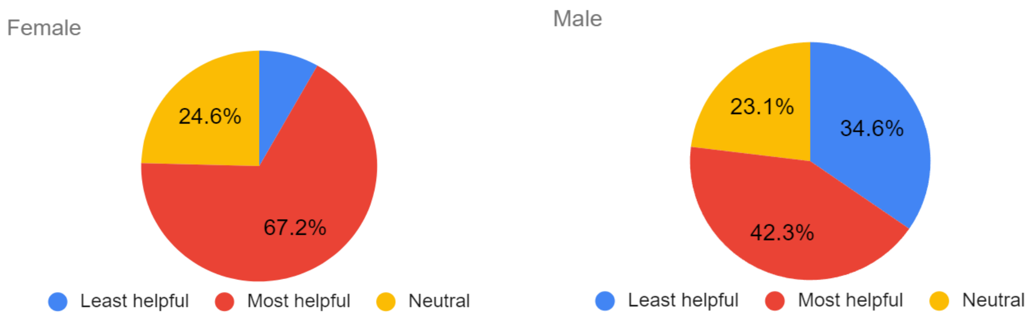


Figure 19. Students' responses to the question "How helpful was using paper blocks in getting started with your project"

RQ2.3: How often did you get to solve a problem or create something with technology?

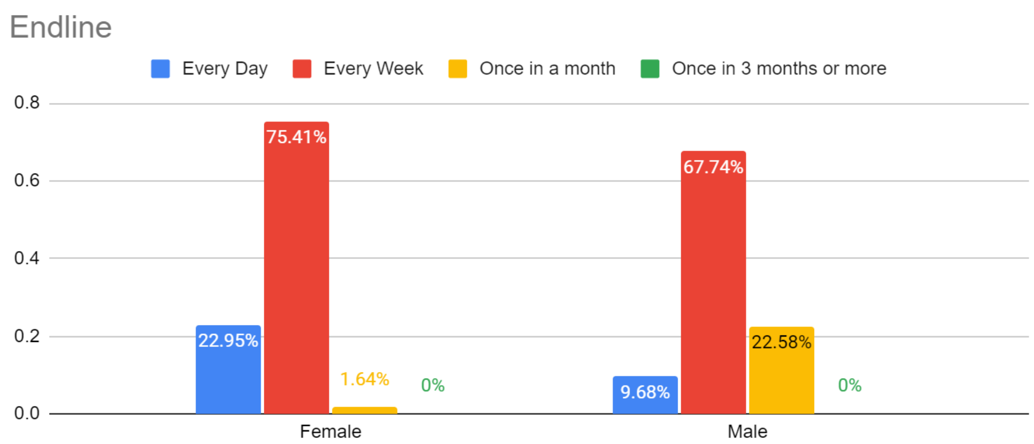
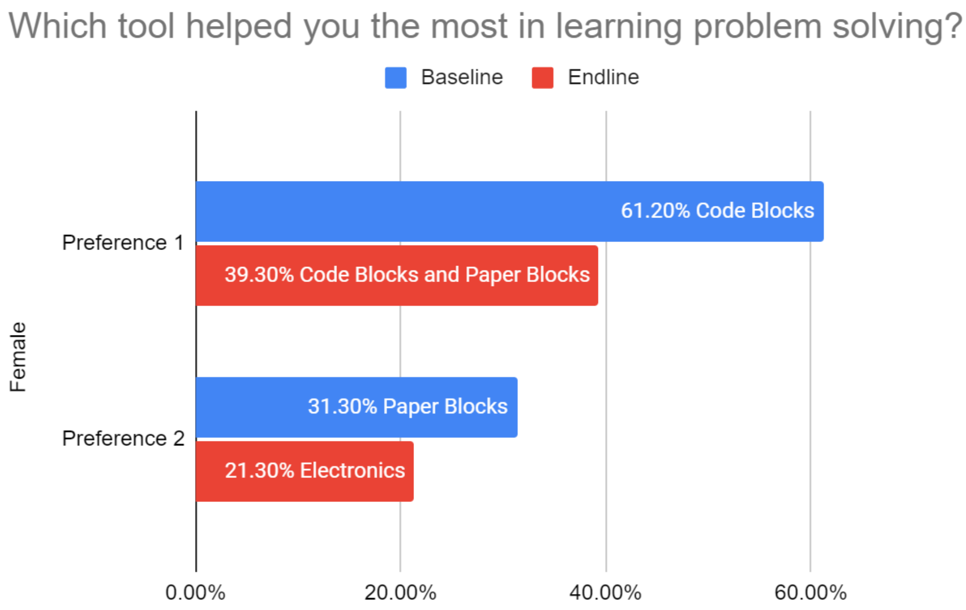


Figure 20.

An average of 88% students reported that they got to solve a problem or create something with technology every day or every week when compared to only around 35% from the baseline

RQ2.4: Which tool was most useful during the intervention in learning skills like problem solving, programming, algorithmic thinking and debugging?



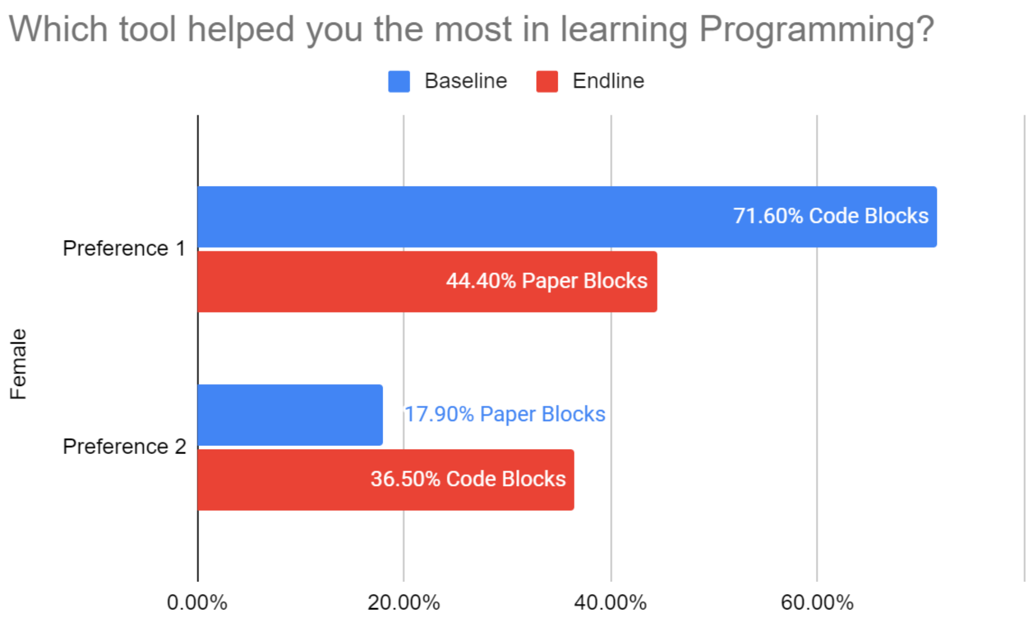
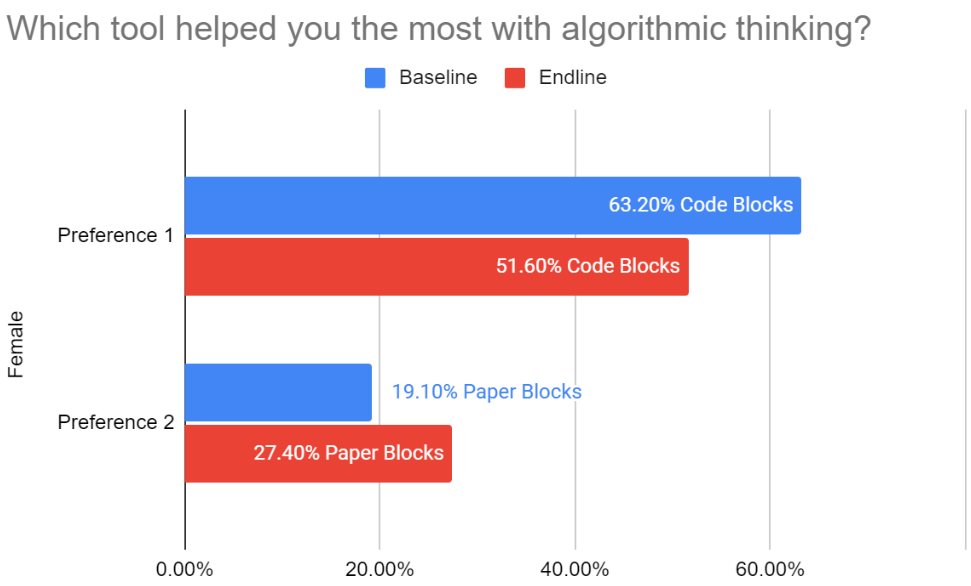


Figure 20 : Tool preference 1 and 2 of female students in learning Problem Solving (Top) and Programming (Bottom)



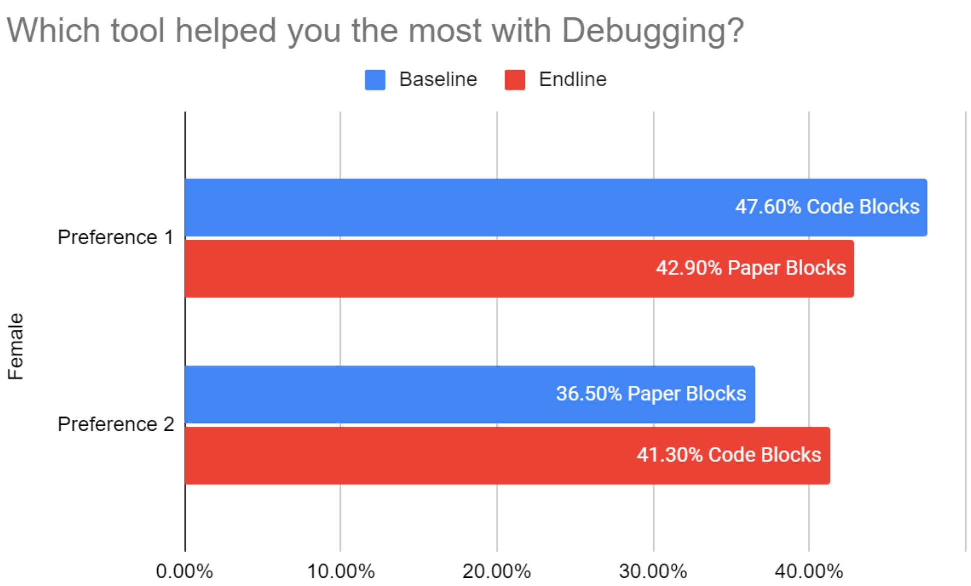
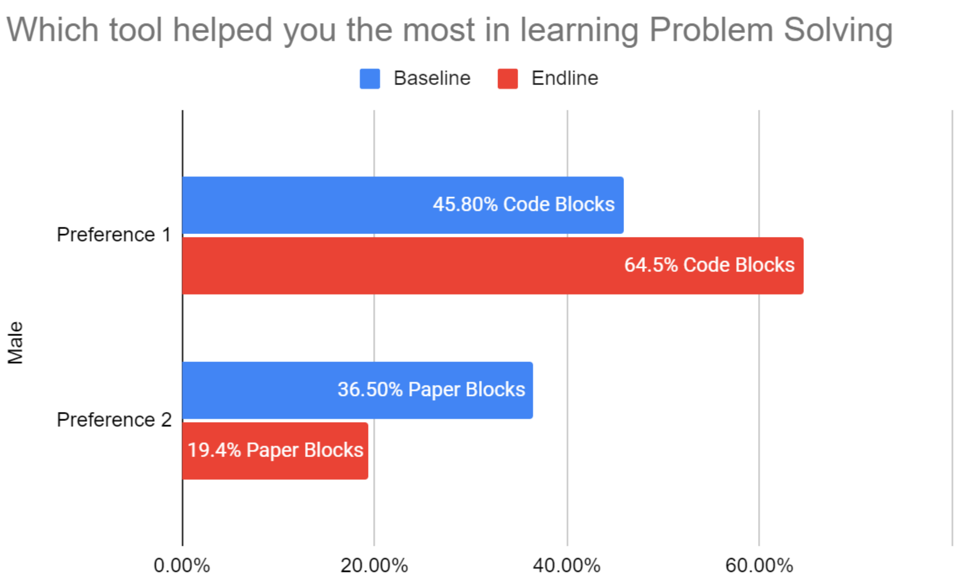


Figure 21 : Tool preference 1 and 2 of female students in learning Algorithmic Thinking (Top) and Debugging (Bottom)

After the intervention, we asked students which tool was most helpful to them in learning specific skills. **Female learners reported paper blocks were the most helpful tool while learning problem solving, programming and debugging**, except for algorithmic thinking (as shown in Figure 20 & 21), where they reported code blocks being the most helpful tool.

There is a 26% improvement in female learners wanting to use paper blocks to learn programming and at the same time a 35% decline in preference to use code blocks to learn programming was observed among female learners.



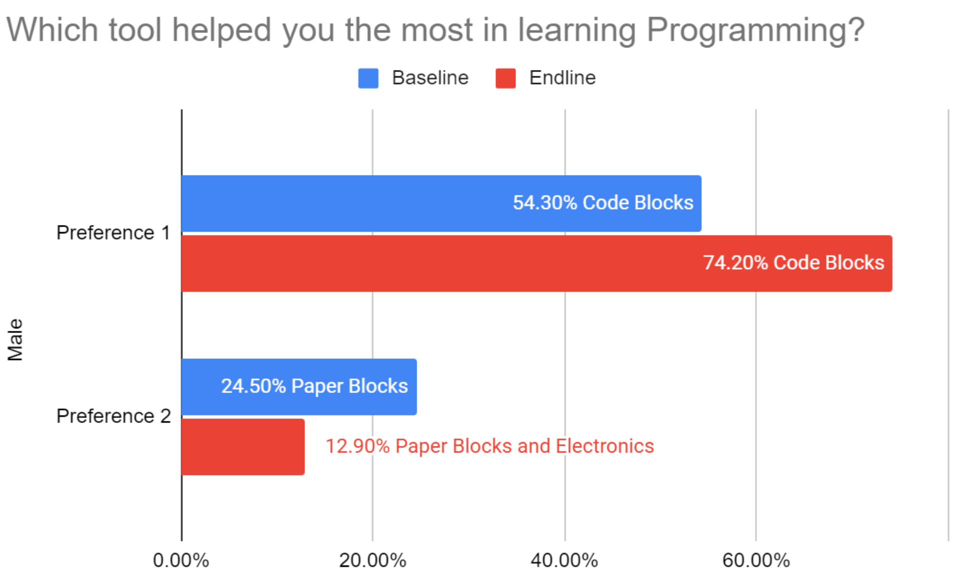
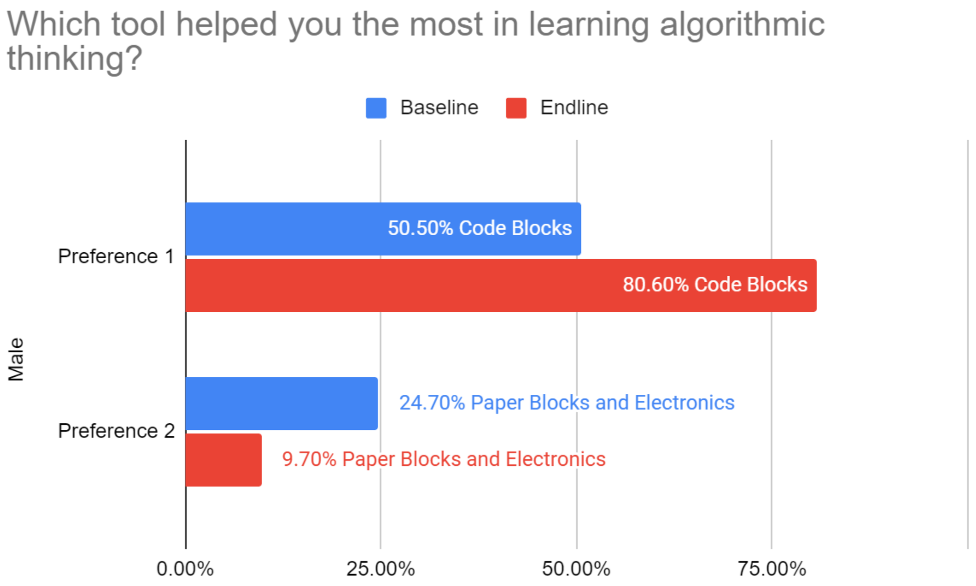


Figure 22: Tool preference 1 and 2 of male students in learning Problem Solving (Top) and Programming (Bottom)

On the other hand, male learners found code blocks to be the most useful tool after the intervention and a subsequent increase in preference to code blocks of more than 20% was observed among male learners.



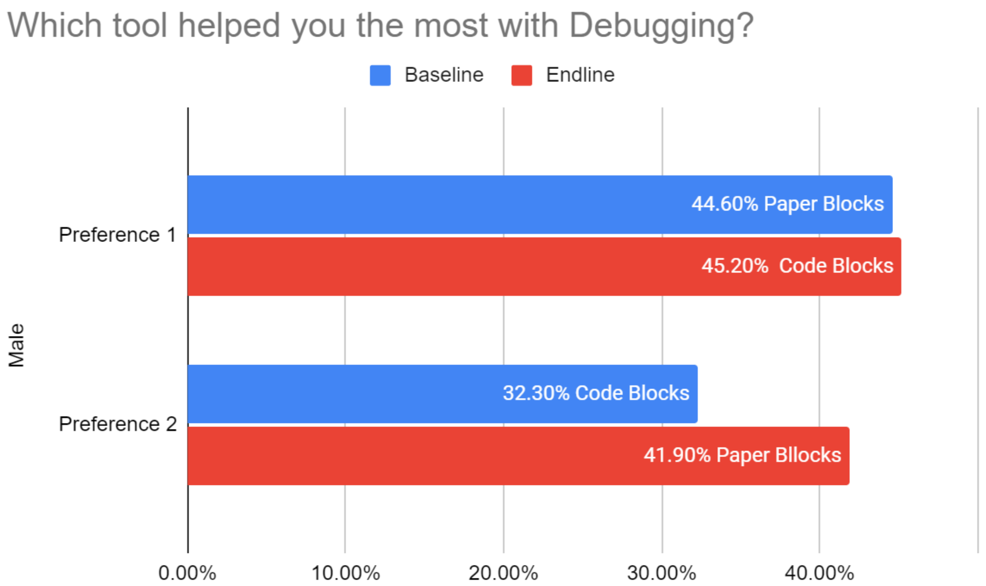


Figure 23: Tool preference 1 and 2 of male students in learning Algorithmic Thinking (Top) and Debugging (Bottom)

Both male and female learners reported paper blocks as a promising and useful tool for learning debugging, i.e finding and reporting errors in their code.

Electronics and microcontrollers were reported to be least useful among the three tools when learning these four above mentioned skills. With close to 97% learners expressing they were most confident in using electronics (as shown in figure 16), the reason for this could be attributed to the fact that electronics on

it's own cannot be a useful tool for learning problem solving, programming, algorithmic thinking or debugging, rather it is a tool that can be used to realize the real-time effects or results of practising these skills, without which students won't be able to see the results of their projects in real-world.
Hence we infer that using electronics in conjunction with code blocks and paper blocks will facilitate learners to learn physical computing effectively.

RQ3 (COMPUTATIONAL THINKING): Does Think Out of the Box support novice students to think computationally?

RQ3.1: Does think-out-of-the-box improve students' algorithmic thinking skills?

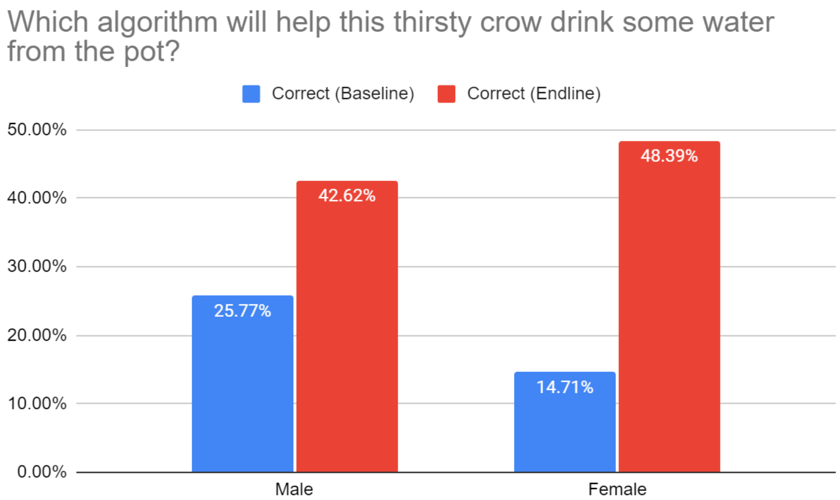


Figure 24. Percentage of students who responded correctly to the question "Which algorithm will help this thirsty crow drink some water from the pot?" across baseline and endline

On an average 45% of students scored correctly on the question of algorithmic thinking when compared to only 20% who got it correct in baseline

RQ3.2: Does think-out-of-the-box improve students' understanding of the concept of repetition?

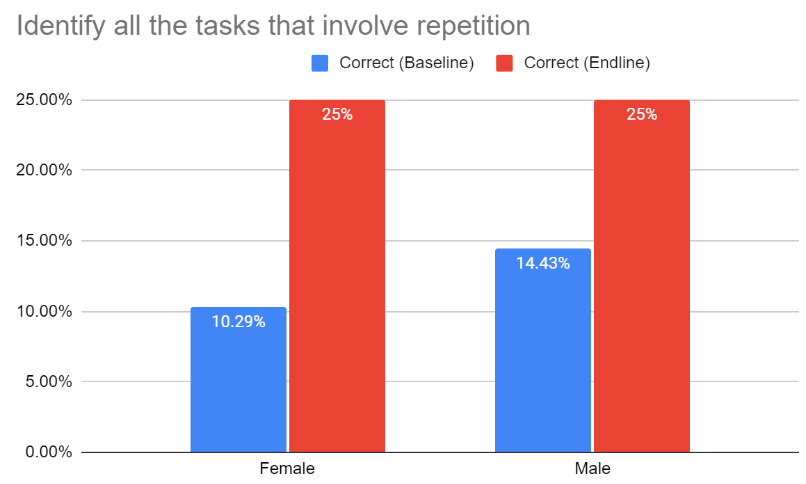


Figure 25. Percentage of students who responded correctly to the question, "Identify all tasks that involve repetition"
25% of students were able to identify and mark all three tasks that involved repetition in them when compared to 12.7% of students from the baseline

RQ3.3: Does think-out-of-the-box improve students' understanding of the concept of sequence?

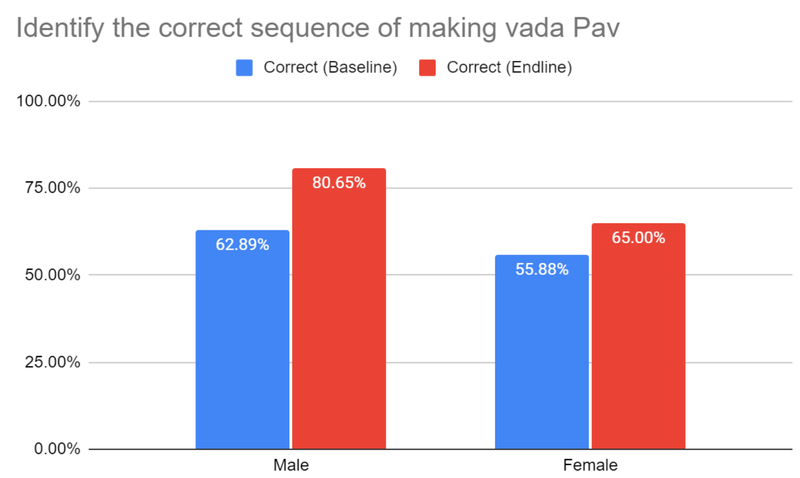


Figure 26. Percentage of students who responded correctly to the question, "Identify the correct sequence of making vada pav"

RQ3.4: Does think-out-of-the-box improve students' attitudes towards making mistakes?

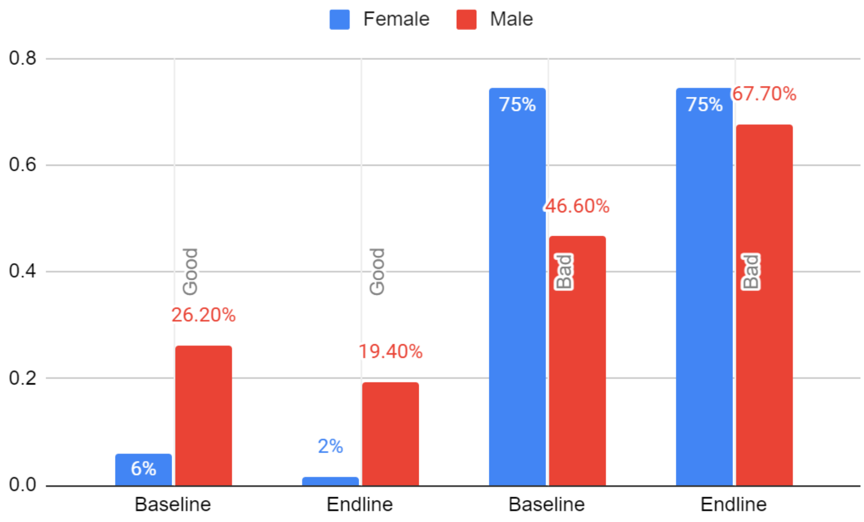
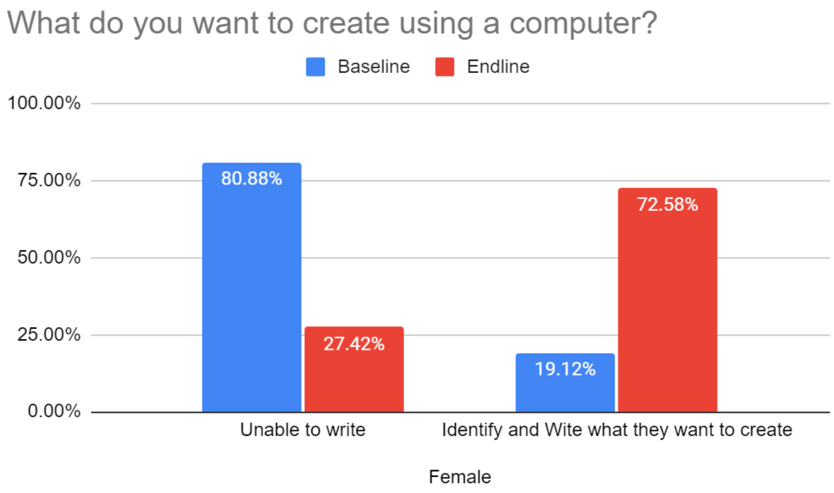


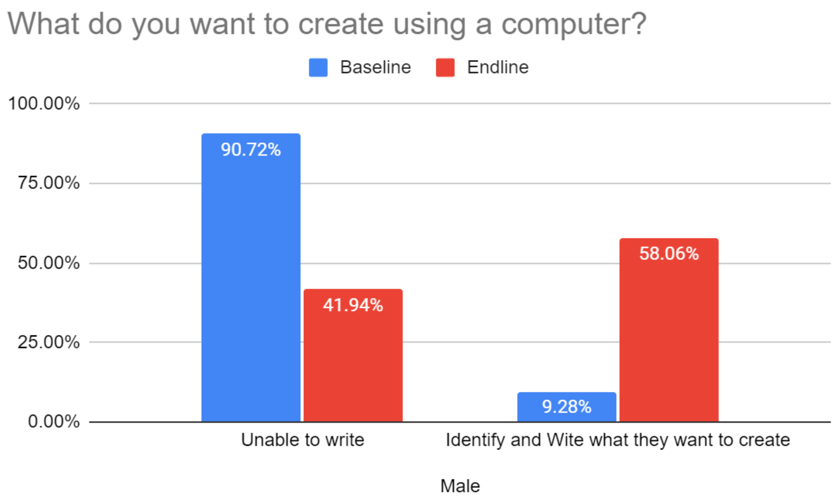
Figure 27: Percentage of students who felt good and bad after making mistakes compared between the endline and baseline

Keeping in mind that computational thinking is not just about skills and concepts, we studied how students' attitudes evolved after the intervention. ***Analysis revealed that our tools and intervention did not have any effect on how students felt after making mistakes.*** *In fact there was an increase in the percentage of students who reported feeling bad. We infer that having the right choice of tools that support debugging and programming is not sufficient on its own to improve students' attitudes towards learning or how they feel when they make a mistake. This analysis will be useful in further investigating the reason why students felt bad when they made mistakes and encourage the culture of appreciating and learning from their mistakes in future interventions.*

RQ4 (PROBLEM SOLVING & CREATIVITY):

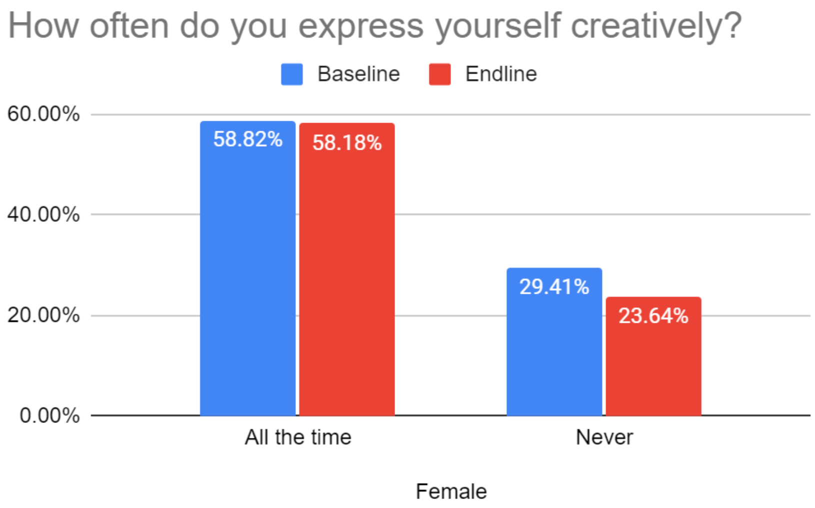
RQ4.1: Does the think out of the box approach support students to think creatively using a computer?

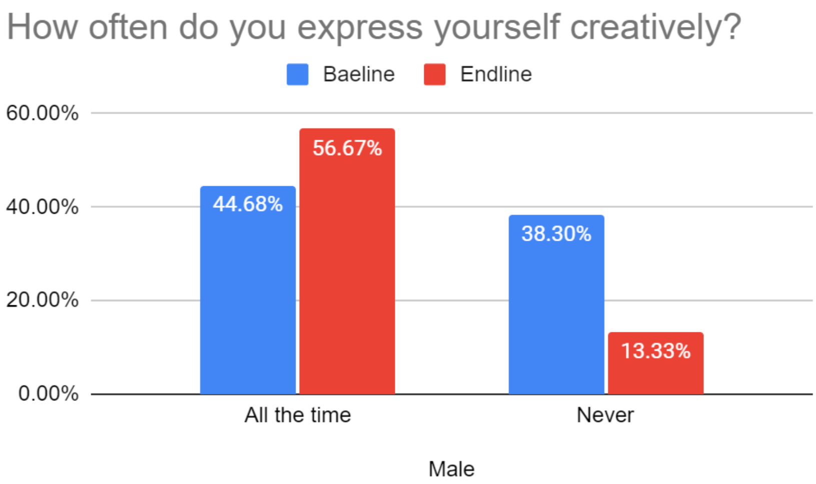




(include graph here)

There is a **53% improvement in female students who are able to write an answer** to the question, **"What do you want to create using a computer?"and 49% improvement in male students who are able to articulate an answer to the same question after the intervention. We infer from the baseline analysis that more girls were naturally inclined to expresour approach could greatly support**





The number of male students who reported that they expressed themselves creatively all the time increased by 12% and those who reported that they never expressed creatively decreased by 25%

REFERENCES:

1. *Garner, Stuart. Reducing the cognitive load on novice programmers. Association for the Advancement of Computing in Education (AACE), 2002.*
2. *Stachel, John, et al. "Managing cognitive load in introductory programming courses: A cognitive aware scaffolding tool." Journal of Integrated Design and Process Science 17.1 (2013): 37-54.*