Research Report for India China Institute

Kolam In Code

By Lavannya Suressh, MFA in Design and Technology, Parsons School of Design

Chapter 1: Introduction

Concept and Significance

Arts and culture have existed as essential daily practices in South Asia for a very long time. In this paper, we will look at *Kolams*, a South Asian art form which has its precedents in mathematical learning; my approach aims to transcend the conventional boundaries of 'traditional' and 'modern' by harnessing the power of programming and code to explore this art form. The complex relationship between the physical and the digital domain has – at least since the dawn of digital manufacturing – been expressed through tools, techniques and technologies associated with their development and deployment. This relationship is less complementary than it is co-dependent: new capabilities in digital fabrication enable the realization of sophisticated tools of design computation, while simultaneously new affordances in design computation inspire innovation in digital fabrication. The level and quality of codependency, whereby design computation enables control and manipulation over the digital domain, as digital fabrication affords control and manipulation over the domain of the physical, determines their level of progress as synergetic design domains. Moreover, the ability to reduce, and even eliminate, the mismatch between digital and physical design tools will ultimately result in fully integrated platforms for design and designers with few 'acts of translation' thereby bridging the gap between the physical and the digital. But, for designers leading the transition from the digital age to the biological age, the challenge to author a cross-domain auxiliary language does not cease here, nor do its associated acts of translation. As we face the challenge to shift between and across fields and their associated units and modes of expression, we must find ways by which to mediate between the physical, the digital and the biological, enabling new ecological perspectives [1]. In this instance, the *Kolam* serves as a traditional artistic practice rooted in the physical realm, providing inspiration for the digital dimension of this project. This endeavor explores a range of concepts, aiming to construct educational frameworks and extend the reach of this art form beyond conventional limits, integrating it into mainstream programming and the realm of art.

It is interesting how an artwork constructed from memory on a daily basis can have such a logical structure with deep conceptual learnings associated with it. *Pulli Kolam* is a ubiquitous art form in south India. It involves drawing a line looped around a collection of dots (*pullis*) placed on a plane such that three mandatory rules are followed: all line orbits should be closed, all dots are encircled and no two lines can overlap over a finite length. The mathematical foundation for this art form has attracted attention over the years [2]. With every sunrise, women wash the floor in front of the houses, and using rice flour, place the dots and draw a *Kolam* largely from memory. The *Kolams* become increasingly complex, with a larger number of dots and more intricate line orbits. Remembering the dot configurations and line orbits is a daily exercise in geometric thinking. The process is immensely pleasurable, especially when a *Kolam* is successfully completed with no loose ends. Throughout history there were gendered exclusions in technologically focused fields and the intelligence required to draw increasingly complex *Kolams* have been undervalued. Exploring the form through technology, mathematical structures and taking it as a cultural precedent to learn mathematical concepts from can play an important role in valuing the contributions to art, science, and technology made by non-western cultures and women.

With our mathematical learnings, it is important to take examples and precedents from our cultures as practices. *Pulli Kolam*, a South Indian art form, is a great way to study fractals, geometry and looping patterns. Approaching this through code helps enshrine the intricacy of practices, which have, for a very long time due to colonialist epistemologies, been seen as 'primitive' or 'primordial'.

I aim to study *Kolams* as a traditional South Indian art form beyond something that's a daily ritual. Taking it as a valued precedent for mathematical learning concepts exploring rules that limit while encouraging creativity at the same time. Approaching this art form with code to create generative code-driven art forms following the same traditional rules and structures show that there is more intelligence required to perform this than is usually credited for and a way of looking at *Kolams* alongside mainstream generative art. Exploring the form through technology, mathematical structures can play an important role valuing contributions to art science and technology from non-western cultures and women.

Chapter 2: Context

Culture and process

Taking a clump of rice flour in a bowl (or a coconut shell), the *Kolam* artist steps onto her freshly washed canvas: the ground at the entrance of her house, or any patch of floor marking an entrypoint. Working swiftly, she takes pinches of rice flour and draws geometric patterns: curved lines, labyrinthine loops around red or white dots, hexagonal fractals, or floral patterns resembling the lotus, a symbol of the goddess of prosperity, Lakshmi, for whom the *Kolam* is drawn as a prayer in illustration. The making of the *Kolam* itself is a performance of supplication. The artist folds her body in half, bending at the waist, stooping to the ground as she fills out her patterns. Many *Kolam* artists see the *Kolam* as an offering to the earth goddess, Bhūdevi, as well.

Kolam means "beauty". The simplest form of the *Kolam* is the *Pulli Kolam* or “dotted *Kolam*”.
6 Dots of rice flour are placed in a grid-like framework, which are then joined to take the form of a symmetrical shape or a regular polygon. Symmetry was of key importance to the *Kolam* artist. It can be compared to fractals in mathematics that can get increasingly complex across scales. These small patterns can be scaled up and interconnected in many ways and the possibilities are exponential. This can be approached with code in 2 ways as we’ll see through the rest of this paper.

The *Kolam* tradition of Tamil Nadu has persisted for hundreds of years and remains a common practice among women in cities as well as in rural areas, among those who are university educated as well as those with less formal education. In recent years, instead of rice, women have often substituted commercially available stone powder, chalk or ink for creating the designs.

Exploring the techniques of skilled *Kolam* artists offers insights into the intricate thought processes behind *Kolams*, loop figures, and the cognitive aspects of comprehending mathematical concepts. I collaborated with Mrs. Muthulakshmi Sankaran from Chennai, a dedicated *Kolam* artist who sees it as a mental exercise. Through our collaboration, she guided me in deconstructing her creative process and the purpose behind each design. I then leveraged these insights to develop a computer algorithm, translating her methods into a detailed, step-by-step programmatic instruction set.

Figure (i) Illustrating the step-by-step process of creating a Kolam by color-coding.

She used different colors to represent each closed-loop drawing within the grid, thus creating a clear visual distinction between each step and process. This use of colors served a functional purpose in understanding the progression and intricacies of her artistry.

By assigning unique colors to specific sections or elements of the *Kolam*, Mrs. Muthulakshmi Sankaran effectively conveyed the sequential nature of her creative process. This technique allowed for a more comprehensive understanding of how each element contributed to the overall design, making it easier to discern the layering and development of the *Kolam* as it evolved step by step.

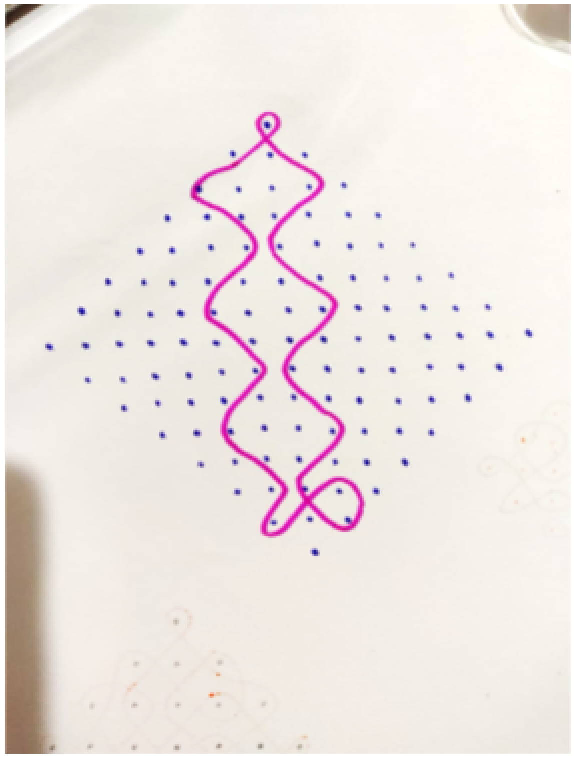


Figure (ii) A single freehand closed loop on a 15x1 dot grid.

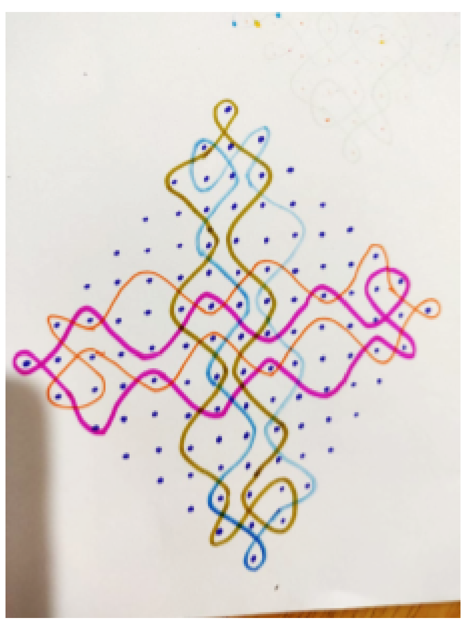


Figure (iii) The freehand loop is replicated four times to form a Kolam, with both 180-degree and 90-degree symmetries.

Her process begins by creating a solitary closed-loop pattern that gracefully traverses the grid diagonally. Subsequently, this singular design is duplicated in a symmetrical fashion across the grid, resulting in a 4-fold symmetry. This approach unveils two noteworthy observations.

Firstly, it highlights the rich variety of movements achievable when traversing diagonally across the grid. This dynamic aspect of her technique showcases the versatility and creativity inherent in the art of *Kolam*-making.

Secondly, this process highlights an interesting restriction: when adding curves around a central dot, there are only three types of approaches available. This constraint reveals the intricate yet organized essence of *Kolam* design, where creativity thrives within specified framework.
In the subsequent section, a thorough examination of *Kolam* rules and constraints will be presented.

Fig(iv) Muthulakshmi Sankaran's Kolam outside their residence on Pongal festival

Chapter 3: Methodology

A Journey into Code

There are a set of rules that follow what is specifically identified as a *Pulli Kolam*. They are as follows:

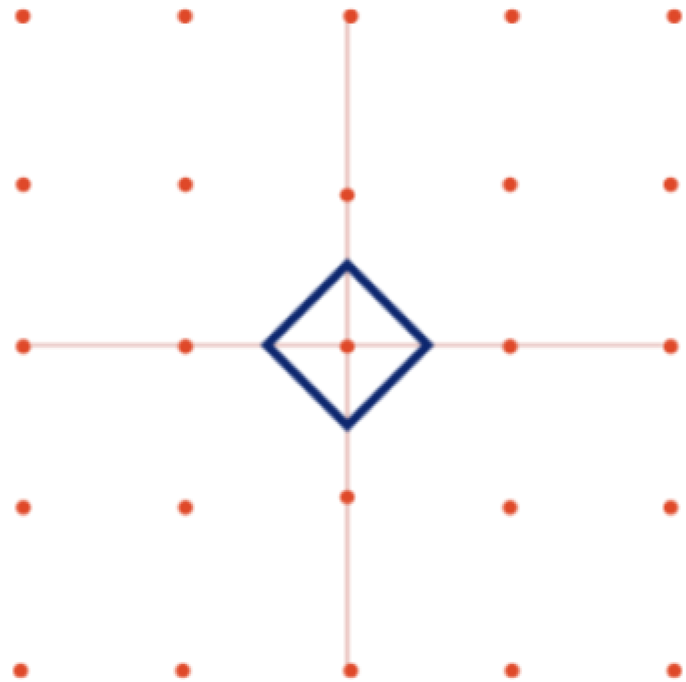
(1) Loop drawing-lines, and never trace a line through the same route.
(2) The drawing is completed when all points are enclosed by a drawing-line.
(3) Straight lines are drawn along the dual grid at an incline of 45°
(4) Arcs are drawn surrounding the points
(5) Smooth drawing-lines. Lines should not bend in a right angle.

The rules and conventions will be explained in further detail going forward.

Method 1: Looping *Kolams* across a grid

In this method, the core physical approach is used in creating a *Kolam*. Dots are placed within a grid-like structure, arranged linearly along a diagonal, serving as the foundational framework. The creation process involves a sequence of loops, incorporating arcs and straight lines, guided by established patterns and principles of symmetry. The size of the grid may be adjusted according to the desired dimensions of the *Kolam*.

To maintain simplicity in this demonstration, a 4-fold symmetry will be employed. This entails initiating a central square (subsequently rotated by 45 degrees), from which the pattern extends and repeats along all four sides.



Fig(v) A grid of dots providing a visual guide for creating a pulli kolam.

The single-line drawing of a *Kolam* consists of three distinct curve types. These curves are defined by the number of adjacent curved sides on a square, which can be one, two, or three as illustrated in Fig(vi). The radius of these curved sides is precisely half the side-length of the square. These numerical values can be assigned to each curve type and processed using a looping function.

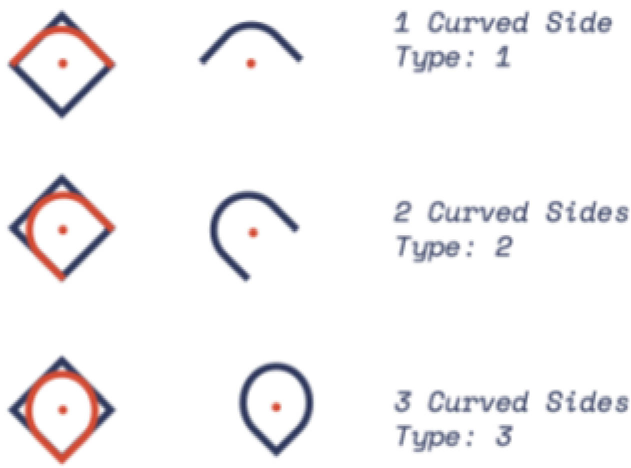


Figure (vi) Three distinct curve types that can be used to create a Kolam

There are a set of rules that we follow while traversing across the grid to make up what distinctively identifies as a *Kolam*. The rules are illustrated in fig(vii) and fig(viii) below showing two case scenarios:

1. If the direction of traversing through the grid is the same
2. If one is traversing to the next row of dots.

The *Kolam* ends with loop type 3 maintaining the rule of leaving no loose ends behind.

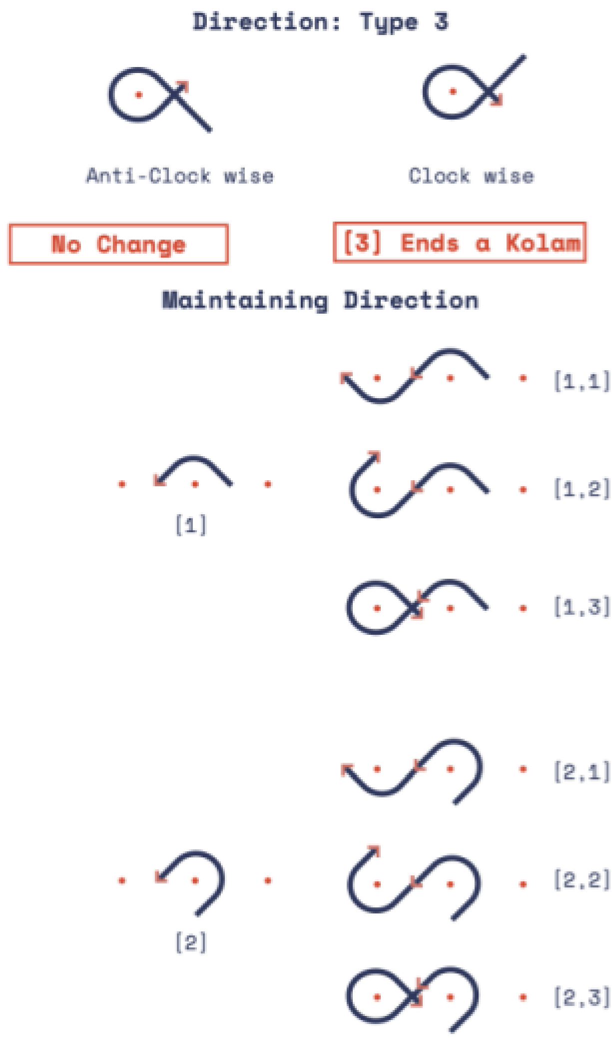


Figure (vii) Transitions among all Kolam curve types maintaining the same direction of motion

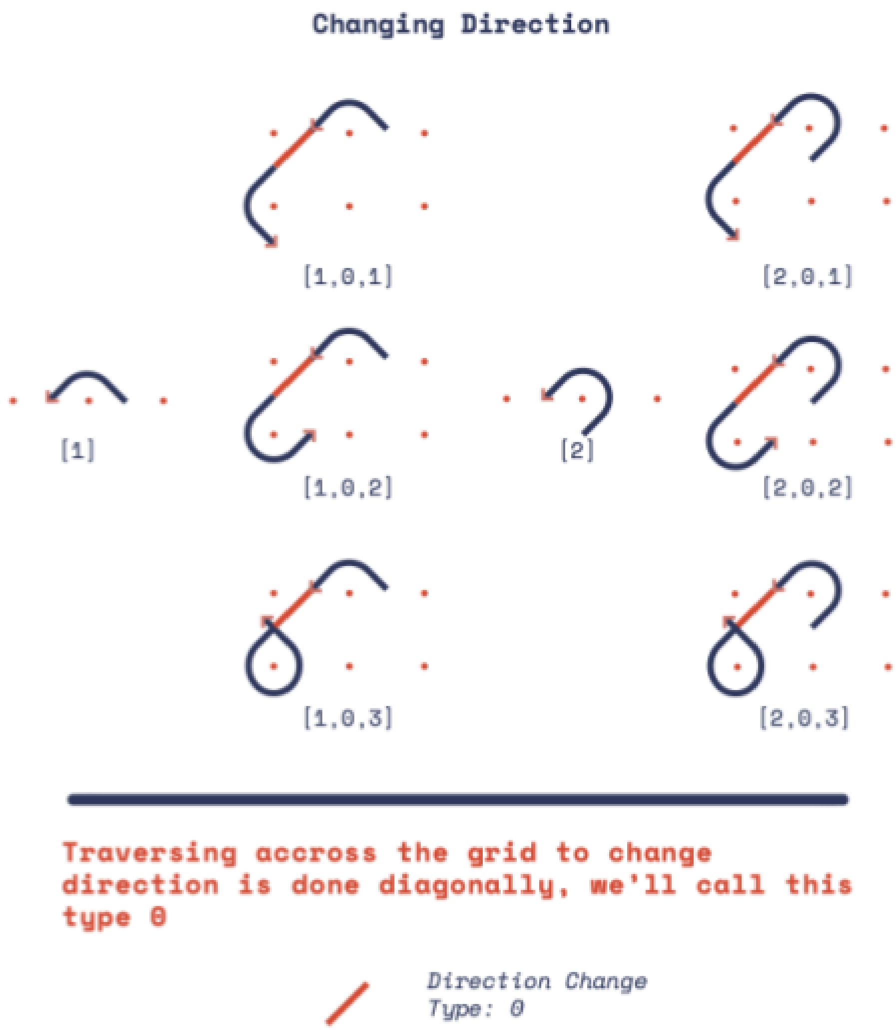


Figure (viii) Transitions among all Kolam curve types when reversing the direction of motion while traversing the grid.

Loop type 0 is a straight line that needs to be drawn diagonally across the dots to continue looping *Kolams* in the next adjacent row of dots. The forward looping is straightforward with numerical values 0,1,2 and 3. This can be inputted into the system in the form of an array.

Since a *Kolam* ends at loop type “3” so the input arrays are limited to types 0,1 and 2 and appended at the end with type 3.

Closing the *Kolam* loop:
We follow a set of rules to loop back after we get to loop type "3" to form the closed loop structure of a *Kolam*. It is illustrated in Fig(ix) below.

1. Loop type 1 connects back with loop type 1.
2. Loop type 2 is reversed back with loop type 0.
3. Loop type 0 is reversed back with loop type 2.

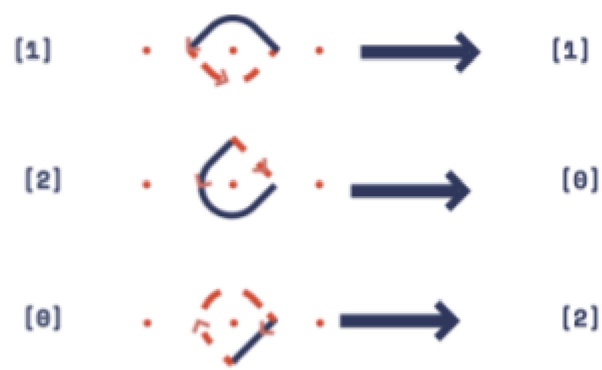


Figure (ix) The closure of a Kolam loop for each curve type.

In Figure (x), we observe the fundamental type of *Kolam* which is denoted as *Kolam[3]*, which serves as the foundation for more complex variants. The complexity of *Kolams* increases with the incorporation of additional curve types to form a closed loop. These straightforward principles are implemented using Turtle graphics in Python, where an array of numbers is the input, and it is translated into a *Kolam* design.

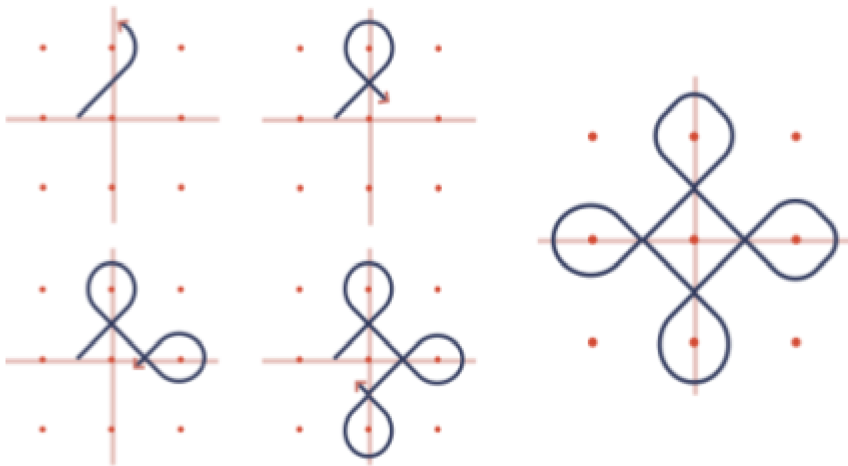


Figure (x) Steps demonstrating the creation process of Kolam[3].

Below are the few examples of Kolams we can derive with the method discussed above.

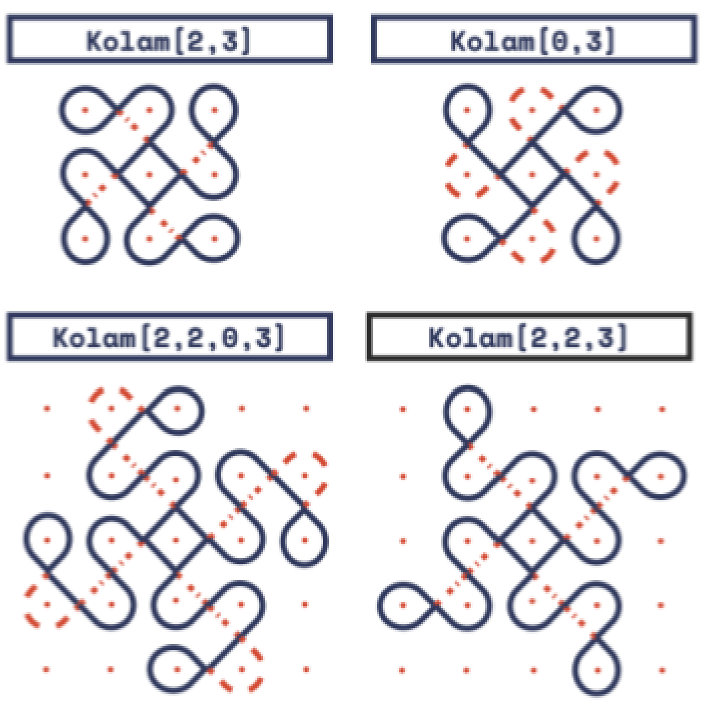


Figure (xi) Kolam examples generated from numerical values corresponding to curve types.

A user-input function guides the traversal around the grid using rectangles with varying combinations of curvature radii. The code then calculates how this path is looped back to form a *Kolam*. This method discusses drawing a *Kolam* with a single continuous line. However, there exist various grid structures, and multiple single loops can be placed to create a *Kolam*. Exploring this aspect in future research could involve establishing rules for the arrangement of these loops, analogous to comparing knots formed by a single rope versus multiple ropes together.

The flowchart depicted in Figure (xii) illustrates the code structure implemented in Python using turtle graphics.

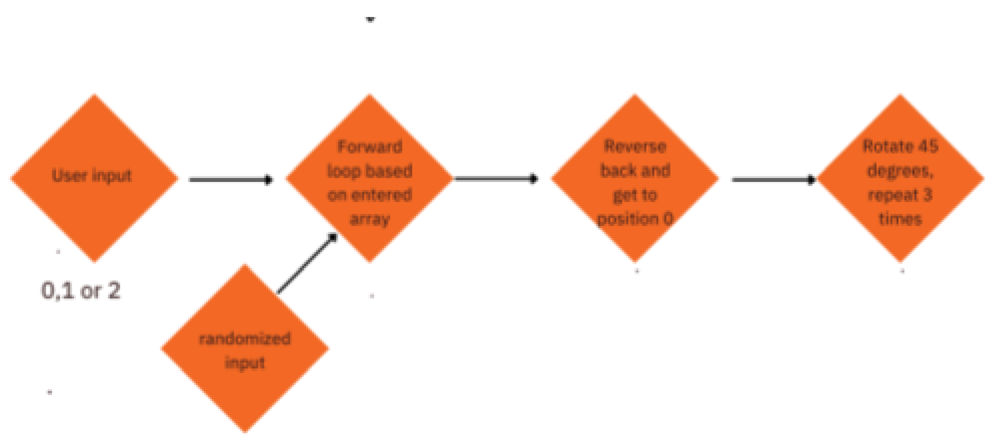


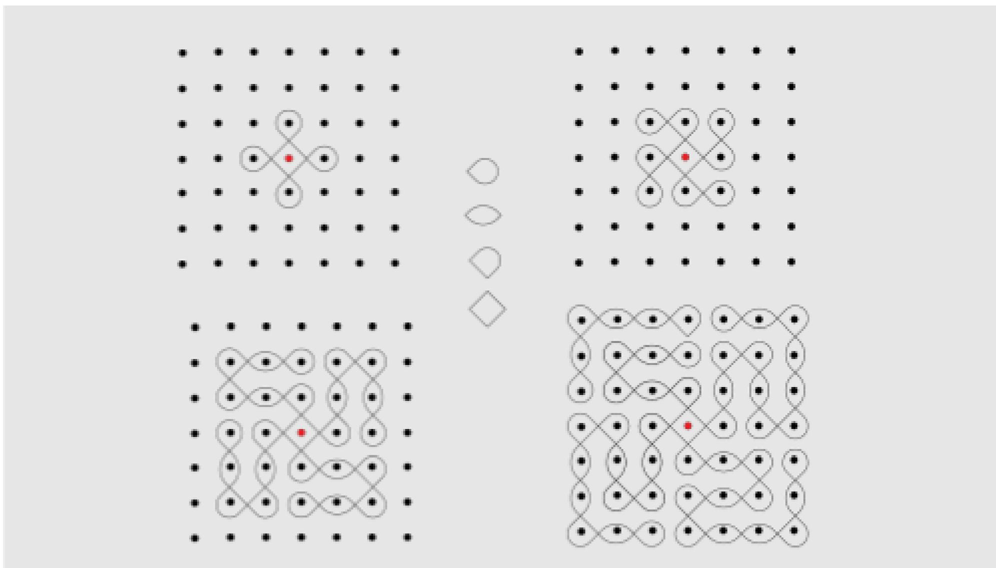
Figure (xii)Streamlined flowchart outlining the processes within the Python code employing turtle graphics.

This method of using code to generate and learn from *Kolams* and with *Kolams* draws similarities to a lot of concepts in mathematics:

1. **Symmetry :** Different dotted grids to create *Kolams* represent different folds of symmetry based on traversing the loops. The dotted grids in *Kolams* are = the maximum number of dots in a row * number of rows with maximum dots* by the minimum number

of dots the grid tapers to. It's important to note that throughout this paper, we have consistently used a fundamental diamond grid as the template for a *Kolam*.

2. **Fractals:** A *Kolam* has the potential to function as a mathematical formula depicting fractals, as derived from the formulation presented above. *Kolams* exhibit self-similarity in patterns that emerge across various scales. Visualizing the dots on an infinite plane, a single Kolam can iteratively scale up. Figure (xiii) illustrates how a basic *Kolam* can scale up across the grid, resembling the characteristic pattern of a fractal.



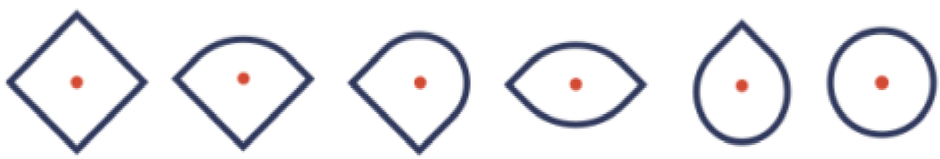
Figure(xiii) Four Kolams illustrating a uniform iterative pattern scaling across a grid, resembling the characteristics of a fractal.

3. **Array grammars:** In this section, we've outlined various approaches to incorporating numerical values into the system, shaping a *Kolam* through the entry of these numbers in array form. The placement of these numerical values on the grid serves as a translation into a *Kolam*. These *Kolam* patterns, interlinked and expandable through multiple iterations, create a structure where numerous closed loops contribute to the formation of a *Kolam*. This exploration can be extended into further research, delving into the concept of array grammar.

Method 2: Kolam Cartesian and Shapes

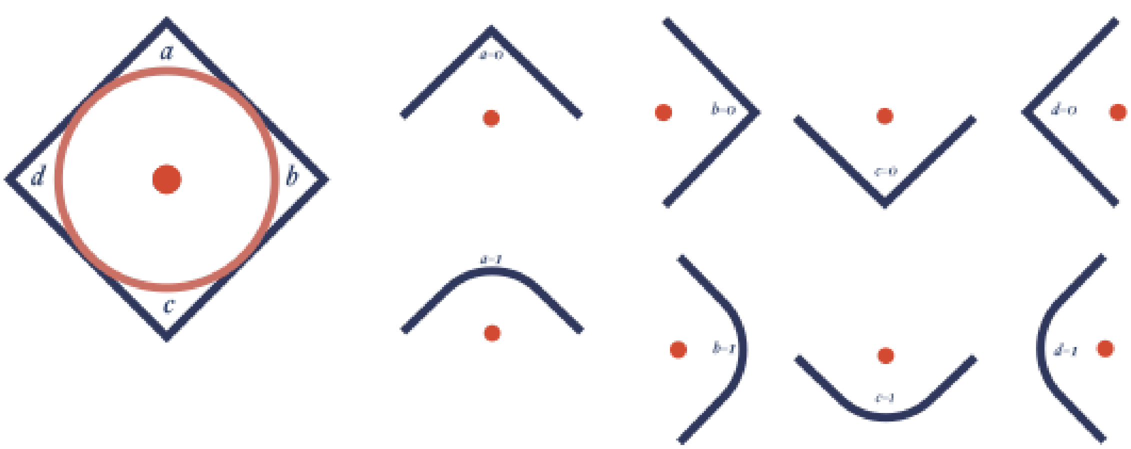
Another way we can look at *Kolams* is based on the shapes that constitute it.This is a method derived from looking at the outlook of what a finished *Kolam* looks like and how it can be represented.

We will visually approach it by placing tiles in a framework. There are 6 types of tiles in different orientations shown in Figure(xiii)



Figure(xiii) 6 types of shapes that make up a Kolam

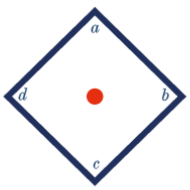
These shapes are based on curved sides of the square in different permutations. Let's denote curved sides as 1 and pointed sides as 0. Taking a clockwise convention, the corners *a,b,c* and *d* can take binary values of 0 and 1.

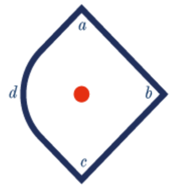


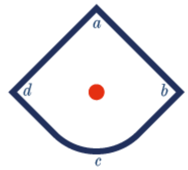
The framework of the grid is similar to a cartesian system but we consider only the coordinates from a diagonal. The framework is illustrated below in fig(vii) .

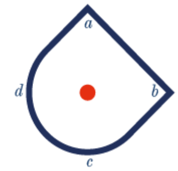
Each side can have two possible values and there are 4 sides so totally there are 2x2x2x2 permutations of shapes. This can be compared to Hexadecimals with 4 bytes of data each taking either a value of 0 or 1.

Figure(viii) Shows all the permutations of shapes.There are 16 shapes that make up a *Kolam* and can be denoted with Hex codes.

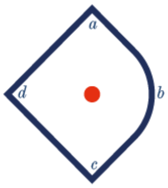


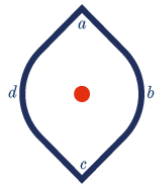


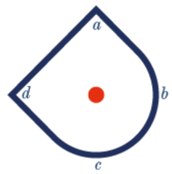


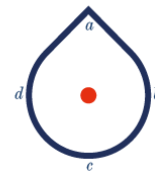


[0,0,0,0]=> 0 [0,0,0,1]=> 1 [0,0,1,0]=> 2 [0,0,1,1]=> 3

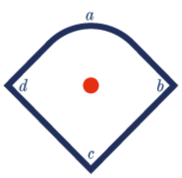


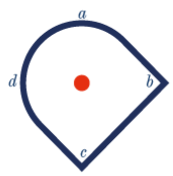


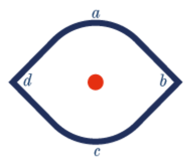


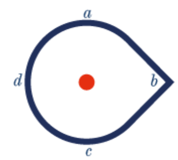


[0,1,0,0]=> 4 [0,1,0,1]=> 5 [0,1,1,0]=> 6 [0,1,1,1]=> 7

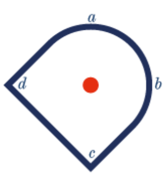


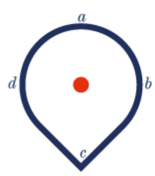


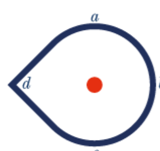


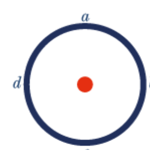


[1,0,0,0]=> 8 [1,0,0,1]=> 9 [1,0,1,0]=> A [1,0,1,1]=> B



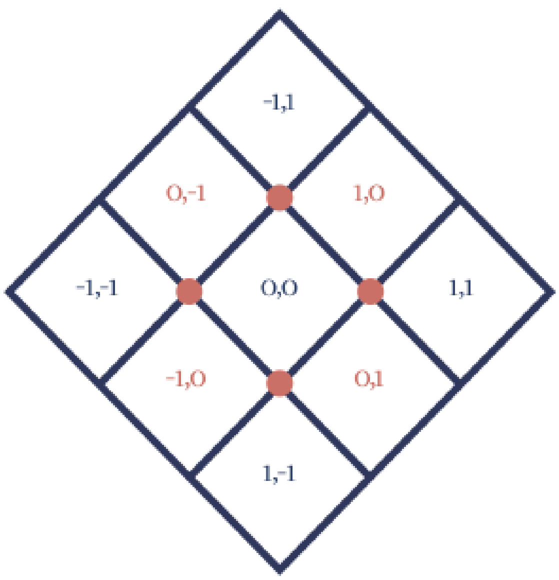




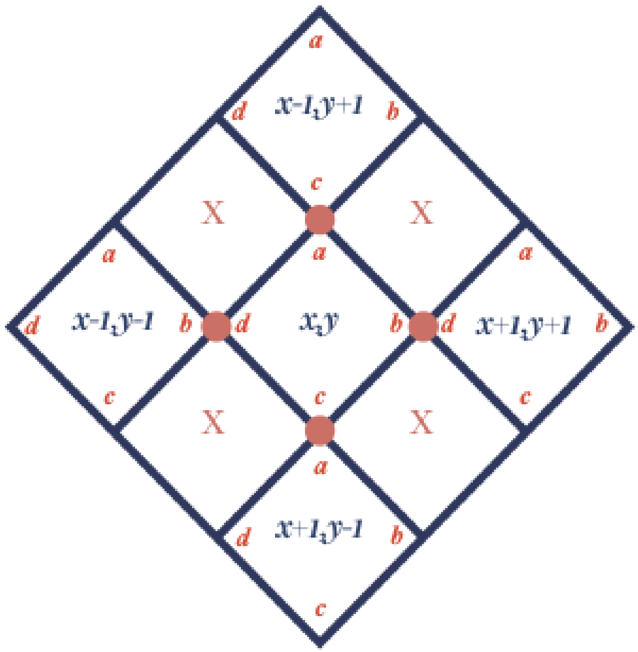


[1,1,0,0]=> C [1,1,0,1]=> D [1,1,1,0]=> E [1,1,1,1]=> F

The grid where we place our *Kolam* shapes is important to understand and represent the *Kolam* structure. It is similar to a simple Cartesian coordinate system in a diagonal space. We can structure a *Kolam* through this with a few simple rules. fig(ix) Shows that grid structure and the corresponding coordinate values. Imagine a regular cartesian system rotated clockwise by 45 degrees. This is the convention we are following here.



We maintain the constant(0,0) space as the point at which the *Kolam* is usually symmetrical around. We will look at how the *Kolam* rules conform around the grid at any given value *x* and *y*. Fig(x) shows the grid structure.



The rules that we follow while constructing a *Kolam* are the following:

1. The sum of the *x* and *y* value should be divisible by 2
2. Adjacent blocks have the same value at the point the joining point to maintain the single line drawing rule
3. Endpoints of *Kolams* must end with a “1”

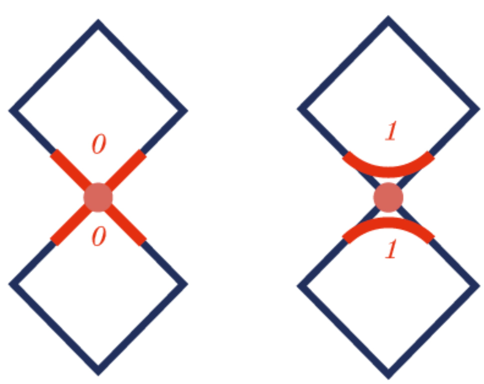
a_{xy} refers to the “a” value of the *(x,y)* coordinate of the tile, The rules are for all the valid tiles around it. They are Illustrated in fig(xi) below:

$$a_{xy} = c_{x-1|y+1}$$

$$b_{xy} = d_{x+1|y+1}$$

$$c_{xy} = a_{x+1|y+1}$$

$$d_{xy} = b_{x-1|y-1}$$

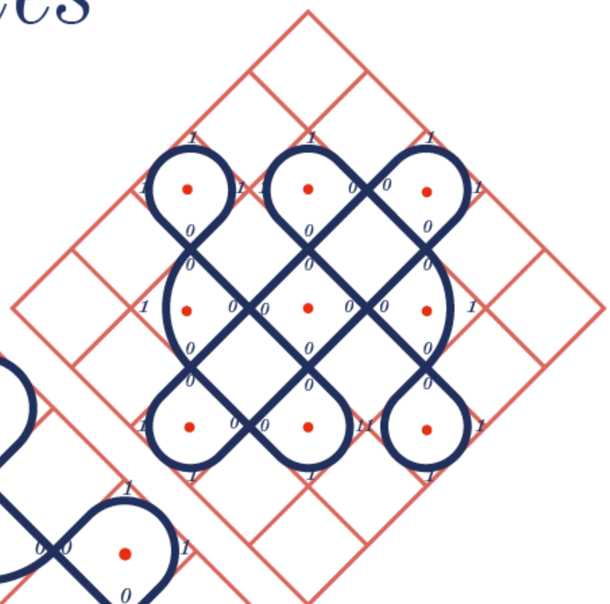


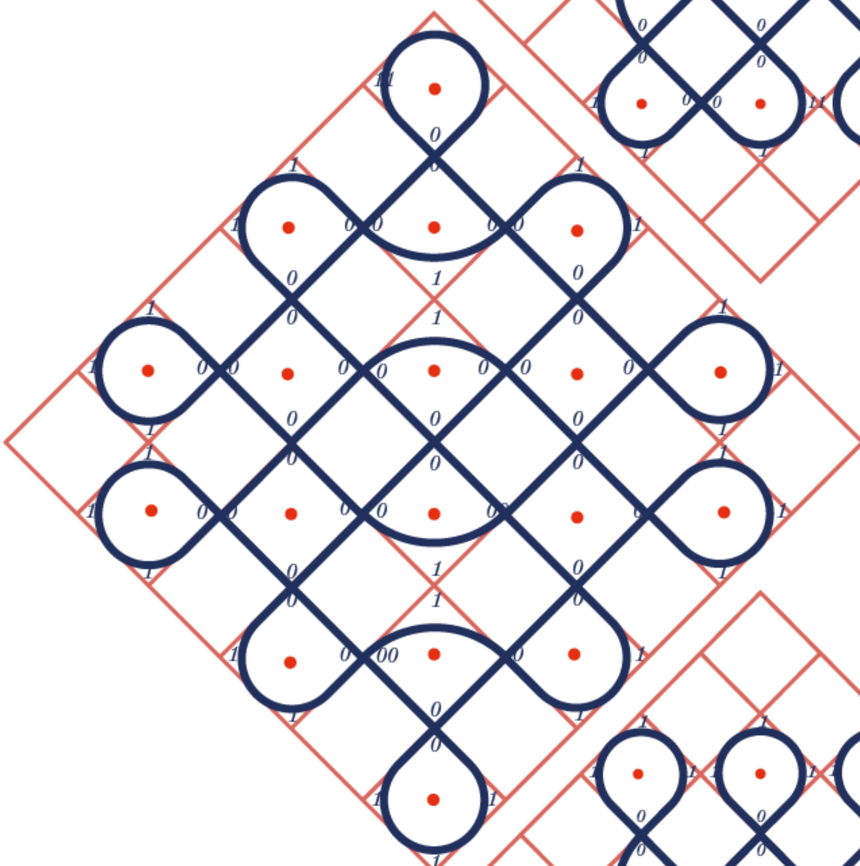
This is akin to data communication in 4-bit computers, where matching bit values are essential to progress in creating a *Kolam*. Figure (xii) illustrates several examples of generated *Kolams*. While this generative approach can be expanded following a set of rules, preserving a degree of symmetry remains an integral part of the aesthetics of a *Kolam*. This method of computing *Kolams* across a space offers significant potential for understanding data and communication.

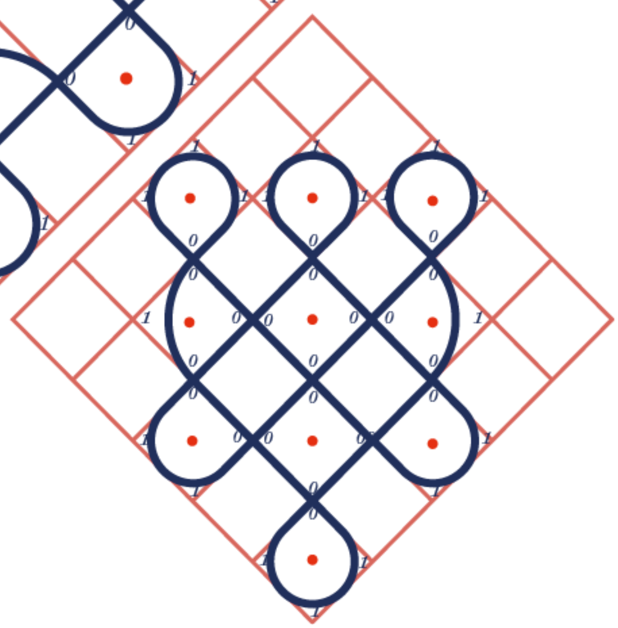
In computer architecture, 4-bit integers and other data units are 4 bits wide. 4-bit CPU and ALU architectures are typically based on registers or data buses of that size. Memory addresses for 4-bit CPUs are often larger than 4 bits, such as 12 bits or more, though they could theoretically be 8-bit. A group of four bits is also referred to as a nibble, with $2^4 = 16$ possible values. Some of the earliest microprocessors had a 4-bit word length and were developed around 1970. While traditional 4-bit computers are now obsolete, recent quantum computers use 4-bit architectures, primarily based on qubits, as seen in the IBM Q Experience.

Kolam is also a continuous line drawing and a space-filling curve in a specifically defined Cartesian system. This not only contributes to near infinite mathematical possibilities but also allows for various orders in the Cartesian system based on the number of dots in a grid that can loop around. It presents an alternative space filled with data types interconnected in continuity.

Examples







Chapter 4: Evaluation

Technology has enabled us to engage with and comprehend a traditional art form. The *Kolam* adheres to a set of conditional statements, leading to the generation of diverse patterns that may be regarded as representations of data or a method of information preservation. The order of a *Kolam* serves as an indicator of its complexity, signifying either the intricacy of its design or the number of loops and dots essential for its construction on a grid. This effort to understand *Kolams* and analyze them with Python and p5 Javascript involves two distinct methods. The Python method focuses on the actual process of creating a *Kolam*, while the Javascript method centers on recognizing patterns in finished *Kolams*. The first approach regards *Kolams* as knots and space-filling curves, while the second pertains to the realm of data and communication. Considering Kolam holistically may enable us to uncover connections between two relatively unexplored mathematical domains.

These mathematical and scientific explanations would not only attract the young generation to practice the art, but would also generate curiosity amongst scholars to preserve the tradition.

It is, however, difficult to distinguish the complexity of *Kolam* at glance. Creating a *Kolam* pattern is expected to be useful for activating/training the brain. It is a wonder that a woman with no math knowledge (except counting) is able to draw any type of complex pattern without much effort. They show their perfection in geometrical presentation, symmetry, straight lines, curves, and so on. Girls capture, encode, and decode the image in their memory with clarity before reproducing it on the ground. *Kolam* can be called an "ethno-mathematical" activity.

The programming languages do not necessarily replicate how the women of Tamil Nadu conceive of and draw the *Kolam* figures. Nevertheless, they underscore the fact that the *Kolam*, and particularly the families of *Kolam*, are more than just a collection of individual pictures; they are unified by systematic procedures and techniques. The fascination of Kolams for computer scientists is not new. There may be no better way to examine an academic construct than to apply it to examples from a tradition and culture outside the one in which the construct arose. But in addition, the computer scientists sought to learn from the figure makers and integrated what they learned into the theory and practice of their own field. This recent phase in the history of the Kolam tradition showcases how mathematical ideas can move beyond their traditional boundaries, interact with an academic endeavor, and in fact contribute to it.

Chapter 5: Conclusion

Kolam is a community-oriented practice deeply ingrained in daily rituals spanning centuries. The genesis of *Kolam*, as well as its varied applications beyond traditional confines, cannot be attributed to any single individual. Examining these contributions on a global scale, especially within the realm of STEM education, where concepts have traditionally been entwined with colonial frameworks, challenges the prevailing Eurocentric paradigms that have long defined the standard.

The effort to decolonize educational frameworks, whether in STEM or design, requires innovative strategies that embrace the wealth of diverse cultural and traditional practices. These alternative approaches can be explored to convey the same foundational concepts, ultimately leading to a decentralized educational structure. Nevertheless, the task of solidifying these concepts as fundamental cornerstones represents a substantial undertaking. This study serves as an initial exploration into the algorithmic understanding of *Kolams* and their documentation, paving the way for their gradual integration into a broader educational dialogue.

Consider the *Kolam* Cartesian, as referenced on page 13, and its relationship to the grid-like structure that defines its form. The Cartesian system is famously named after the French mathematician and philosopher René Descartes, who introduced this concept in 1637 during his residency in the Netherlands. It's noteworthy that this system was independently discovered by Pierre de Fermat, who also delved into three-dimensional mathematics. *Kolams*, on the other hand, uses a framework strikingly similar to the Cartesian coordinate system, evoking the grid-like structure formed by dots for space-filling curves. Notably, *Kolams* have existed for centuries, pre-dating the establishment of the Cartesian system.

Another point of reference for comprehending two-dimension *Kolams* is found in the realm of Euclidean geometry, a field that bears the name of the ancient Greek mathematician Euclid. His *Elements* serves as a foundational work in geometry, offering a systematic and logical exposition of the subject. Similarly, Hilbert's space, named after the renowned mathematician David Hilbert, is a fundamental concept in functional analysis. Hilbert's spaces play a pivotal role in various mathematical disciplines, including quantum mechanics and signal processing.

These are just a few examples among a plethora of mathematical ideas that are firmly associated with the names of their originators. Notable figures such as Pythagoras, Fermat, and Euler have also left indelible marks in mathematical history through their pioneering contributions to number theory, probability, and graph theory, respectively.

However, it is essential to recognize that while these celebrated figures have played a pivotal role in the development and formalization of mathematical concepts, mathematics as a whole is a collaborative and evolving endeavor. Significantly, several historical cultures have cultivated a rich repository of mathematical traditions, which are distinctly characterized by their communal orientation and have their origins within marginalized communities. These mathematical traditions encompass a spectrum of practices, *Keffiyeh* patterns from Palestine, *Kolams* originating in South India, Navajo weaving from Native American traditions, Islamic geometric patterns rooted in Islamic art and architecture, and African Kente Cloth, traditionally woven by various African communities.This recognition holds particular relevance for those students who have historically experienced disproportionate underrepresentation within STEM career pathways, influenced by various determinants such as racial disparities, socioeconomic factors, and gender imbalances.

Moreover, the attribution of mathematical concepts to specific individuals can sometimes overshadow the collective nature of mathematical discovery. It is noteworthy that these mathematical traditions have taken root independent of formal mathematical education or academic foundations. Therefore, it is imperative to underscore the critical importance of recognizing and not undervaluing these contributions. Such recognition holds profound implications for broadening the accessibility of education, particularly within the domain of STEM, which has historically exhibited traits of exclusion and male domination. Furthermore, these fields have largely adhered to Eurocentric conventions, often featuring nomenclature that celebrates individuals of predominantly European descent, notably those of Caucasian heritage.

Furthermore, there exists a prevalent tendency to trivialize these traditional art forms by categorizing them as mere "crafts" or portraying them as romanticized handmade art objects designed for the consumption of a predominantly Western audience. In truth, these cultural expressions transcend their physical manifestations, assuming the role of foundational constructs upon which a multitude of diverse concepts are built.

Mathematical knowledge is an intricate web of interrelated ideas and theorems, with numerous threads of influence and inspiration extending far beyond the confines of individual action. The collaborative, multicultural, and cumulative nature of mathematics underscores its depth and richness, demonstrating that the field is much more than a collection of individual achievements. It is a collective human endeavor, constantly expanding and deepening our understanding of the fundamental principles that underlie the universe.

References:

1. Bader, C. (2014). Translational Design Computation. Bachelor of Science in Applied Computer Science, University of Applied Sciences Ravensburg-Weingarten. Retrieved from https://dspace.mit.edu/handle/1721.1/112913?show=full

2. Gopalan, V., & VanLeeuwen, B. K. (Year, if available). Title of the paper. In Materials Science and Engineering, Pennsylvania State University, University Park, PA 16802, USA. Retrieved from https://arxiv.org/pdf/1503.02130.pdf

3. "Indian Rice Art - Kolam." Atlas Obscura. [URL: https://www.atlasobscura.com/articles/indian-rice-art-Kolam]

4. Ascher, Maria. "The Kolam Tradition."

5. Yanagisawa, Kiwamu, and Nagata, Shojiro. "Fundamental Study on Design System of Kolam Pattern."

6. "Mathlapse: Math Art in South Indian Traditional Art - Suzhi Kolam." Imaginary. [URL: https://www.imaginary.org/film/mathlapse-math-art-south-indian-traditional-art-suzhi-Kolam]

7. "4-Bit Computing." Google Arts and Culture. [URL: https://artsandculture.google.com/entity/4-bit-computing/m02k_1n?hl=en]