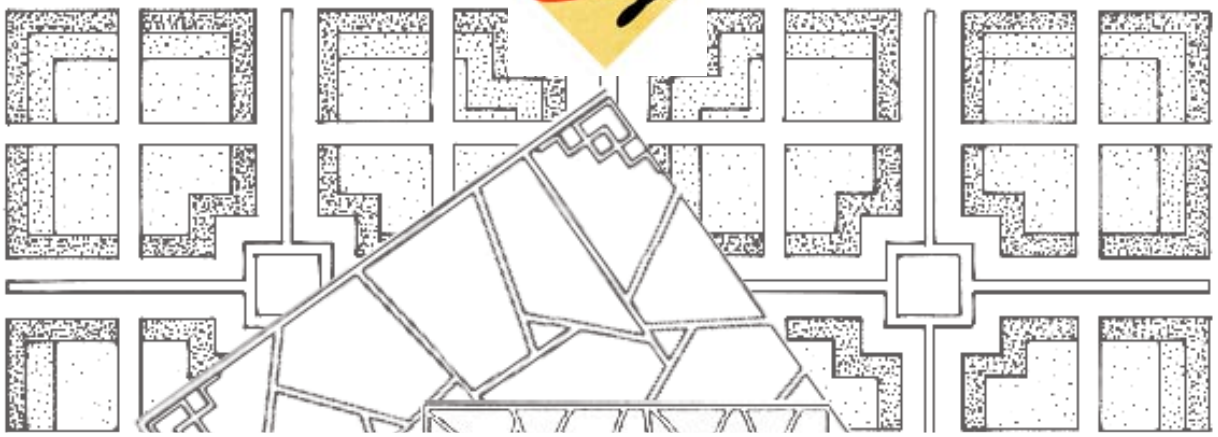
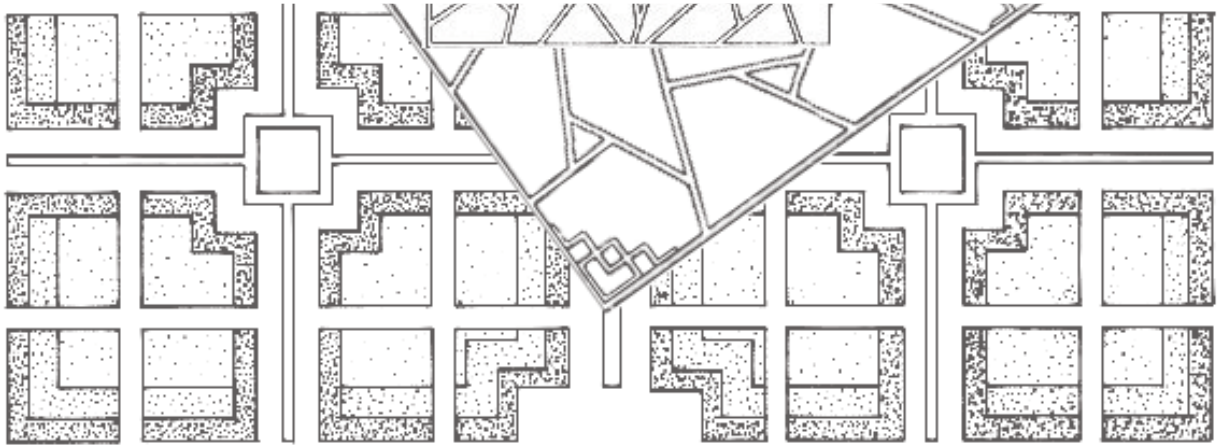


Shape Grammar

Creating forms using different shapes



Guided By
Mamata N Rao

Submitted By
Shivendra Singh
Design for Digital Experience
NID R&D Campus, Bangalore

Project Overview

Shape grammar is the process of shape rules and generating engines to generate different forms. The concept arrives as some time our brains will think in a fixed direction so that we come up with similar kind of forms using the shapes.

Using computer as a tool we can generate different forms using shapes, shapes rules and generating engines. The main goal of this project to give the user a tool that can generate different forms in no time, and helping the designer & architect society.

Objective

To give the society a tool which can use the shape, shape rules, and some compilers to come up with different forms.

Shape grammars are well-suited for teaching composition and visual correlates such as proportion and symmetry.

With a shape grammar approach, one may begin to answer the question “Where do designs come from?” by pointing to rules that generate them.

Just like any other creative process, the design of rules involves intelligence and discipline on the one hand and intuition, imagination and guesswork on the other.

Introduction

A shape grammar is a set of shape rules that apply in a step by-step way to generate a set, or language, of designs. Shape grammars are both descriptive and generative. The rules of a shape grammar generate or compute designs, and the rules themselves are descriptions of the forms of the generated designs.

Shape grammars have properties aimed at making them especially suitable for designing, without sacrificing formal rigor.

First, the components of shape rules are shapes: points, lines, planes, or volumes. Shape rules generate designs using the shape operations of addition and subtraction, and spatial transformations familiar to designers such as shifting, mirroring, and rotating. In short, shape grammars are spatial, rather than textual or symbolic, algorithms.

Second, shape grammars treat shapes as non atomic entities--they can be freely decomposed and recomposed at the discretion of the designer. This liberty allows for emergence--a feature that distinguishes shape grammars from set grammars, the most common kind of formal grammar. Emergence is the ability to recognise and, more importantly, to operate on shapes that are not predefined in a grammar but emerge, or are formed, from any parts of shapes generated through rule applications.

Third, shape grammars are nondeterministic. The user of a shape grammar may have many choices of rules, and ways to apply them, in each step of a computation. As a design is computed, there may be multiple futures for it that respond differently to emergent properties, or to other conditions or goals.

The creative abilities, whatever they may be, are a part of grammatical design, the value of rules as opposed to more conventional approaches, particularly for design education. Basically, the value of using rules to design is two-fold.

First, rules make explicit or externalise a user’s design ideas so that they can be examined, changed, and communicated more readily.

Second, rules make possible multiple design solutions rather than a single solution. However, it is not the multiple solutions themselves that are important. Rather, it is the possibility of *choosing* between different solutions that is important. The process of evaluating and selecting among different designs again brings into focus a user’s design intentions.

Analysis-driven approaches to grammatical design have additional benefits. With these approaches, multiple skills are taught in a coordinated way. User learns about the work of accomplished

designers or their own work in progress, about ways of designing, and about ways of developing their own work.

Pure analysis applications have much educational potential. There is no better way to learn about styles or languages of designs (at least compositionally) than by either studying shape grammars already written for languages or by writing grammars oneself. Good analytic grammars are both parsimonious and descriptive. They are eye-openers, revealing simplicity or regularities behind designs seemingly complex or random. They reveal the thoughtfulness, behind designs that user might otherwise take as unfathomable.

Analytic grammars also embody general design strategies that users can learn from and use in their own work. Different grammars for very different languages (temporally, culturally, geographically) often use common design strategies. For example, in a number of shape grammars, designs are based on an abstract grid or parti. Spaces are delineated within the grid, and then finer details are added within these spaces. The Palladian grammar, the Japanese tearoom grammar, and Li's Yingzao Fashi grammar all work in this fashion. A number of grammars use subdivision as the basis for designs. This strategy is useful when designs in a language have the same, regular boundary. The ice-ray grammar, the Hepplewhite chairback grammar, the Siza grammar, all of the painting grammars (Vantongerloo, Glarner, and Diebenkorn), and to some degree, the bungalows of Buffalo grammar, work in this way. Other grammars use an additive process for generating designs. This strategy is useful when designs in a language have irregular or diverse kinds of boundaries. With this approach, designs are generated beginning with one part (the core) of a design to which other parts are successively added. The Wright grammar, the Queen Anne grammar, and Colakoglu's Bosnian house grammar follow this approach.

Implementation

Computer implementations are good demonstration tools for showing novices the range and power of shape grammars. They can allow students and designers who do not wish to deal with the technicalities of grammars, to develop or use shape grammars with success. For advanced shape grammarians, who understand how shape grammars work, they allow for rapid explorations of rules and design possibilities. Shape grammars are powerful devices and the power of computers is needed to explore their limits.

One critical problem in authoring an original grammar is how to develop a grammar that meets the goals and constraints of a particular project. A commonly asked question by design students writing grammars for a project is "How should I start?". At some point in the process of developing a grammar--if not at the start--a connection must be made between rules that describe spatial form, and the goals of a project that may describe anything from function to meaning to aesthetics and so on. Making this connection is not an easy task because shape grammars are in general unpredictable.

The applications created for implementing shape grammars usage, Using one shape, i.e. Circle. The one circle is used to generate different form by manipulating the value using mouse pointer.

Figure 1: The Creation of Form from Circle

The mouse pointer is moved in the application to generate different forms using one shape, and when the mouse pointer is idle the shape is a circle.

First Shape is when the mouse pointer is idle, Second Shape is when the mouse pointer is moved outside, Third Shape is when the mouse pointer is moved inside, Fourth Shape is when the mouse is left idle and the shape is coming back to the original form i.e. Circle.

The Second Application is also using one shape, which is circle but we can control the Size, Rings, Alpha, and number of circle. The generated forms can be saved for further use in different project.

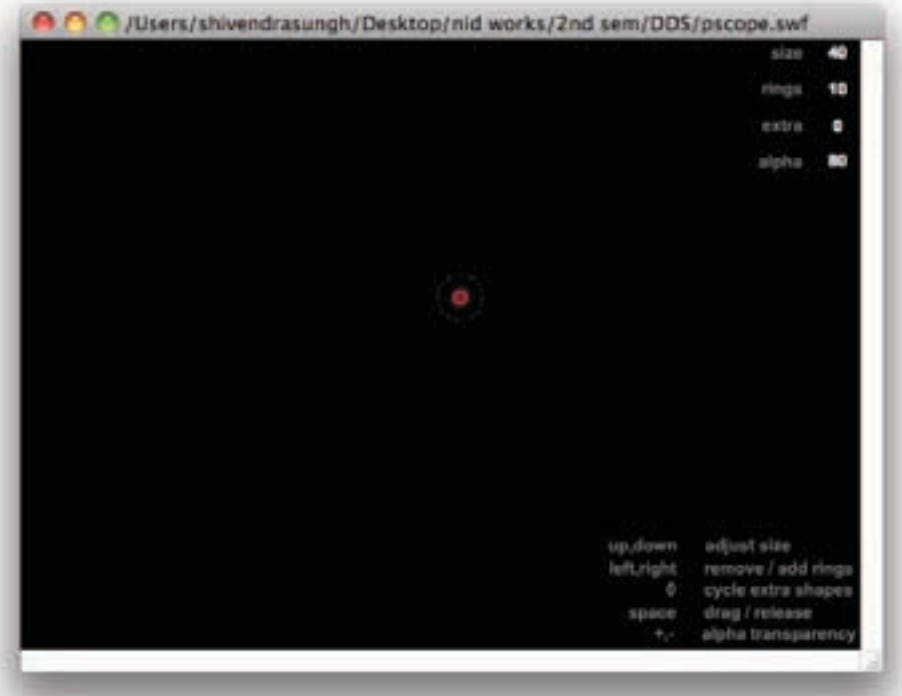


Figure 2: The Application to create forms using circle

Figure 3: The Creation of Form from Circle with user manipulation

First Shape is when the mouse pointer is idle, Second Shape is when the Size is increased, Third Shape is when the Rings are increased, Fourth Shape is when the Size and Rings are increased outside,

The Third Application is for learning the Shape Grammar rules.

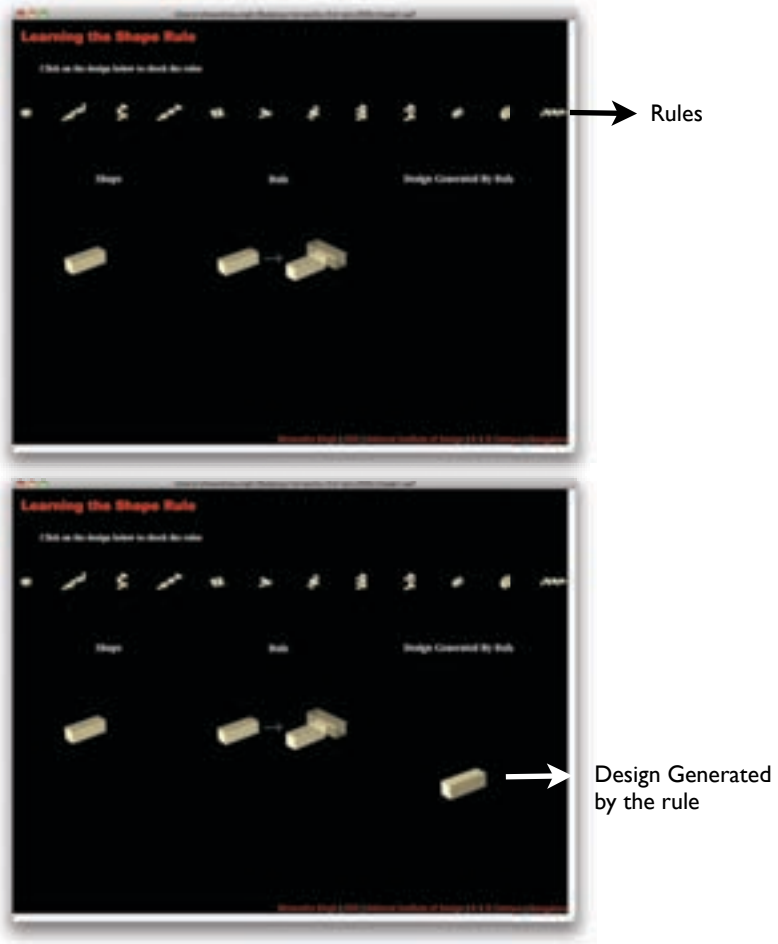


Figure 4: Learning Shape Rules

The Shape Rule can be learnt using this application, the above image display step by step one rule. We can also learn different rules using the above application by clicking on the rules to know about them.

The Fourth Application is using a Game, Using one shape and one rule, how can we generate different shapes, and we can use generated shape for different projects.

First the concept is generated, then made a paper prototype of the game and tested with users. The Shape and Rules are defined.

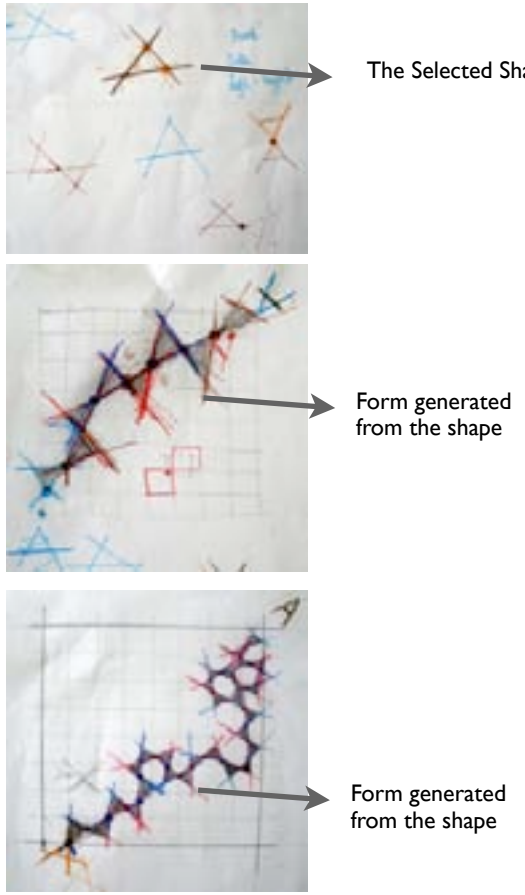


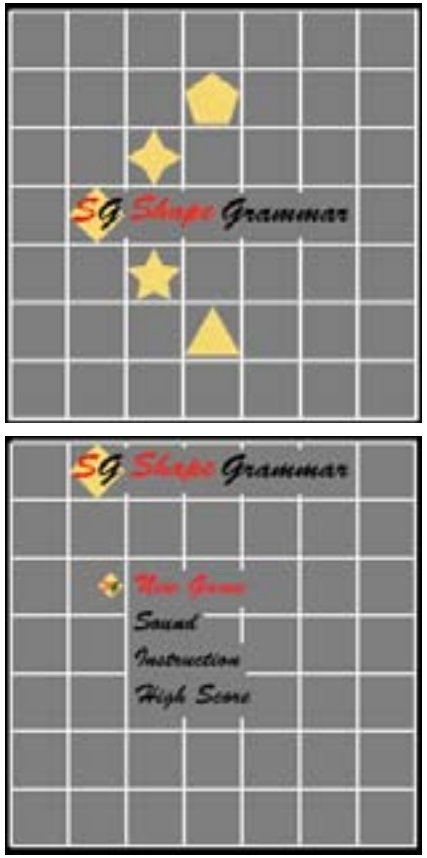
Figure 5: Shape Game - Paper Prototype

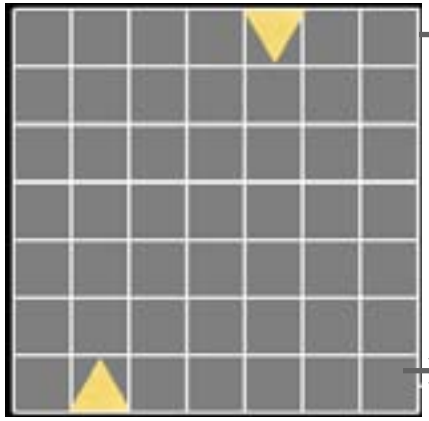
The Rules for the games are as as follows:

- The Shape and the rule's are pre defined in each level.
- The final shape placement & orientation is decided by the system.
- The User can select the place of placement of the shape.
- The User can also select the first orientation of the shape.
- The User can select where he wants to place the flipped shape.
- The user has to select the point, and then press ok to put the flipped shape.

Prototype

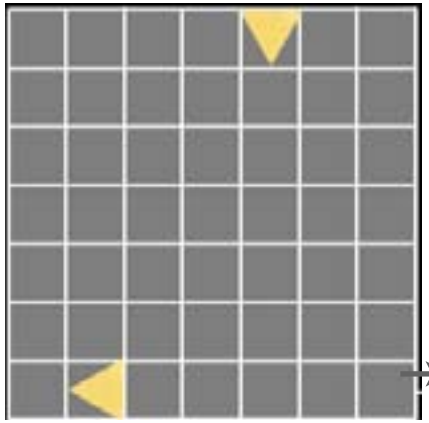
The Prototype of the game.



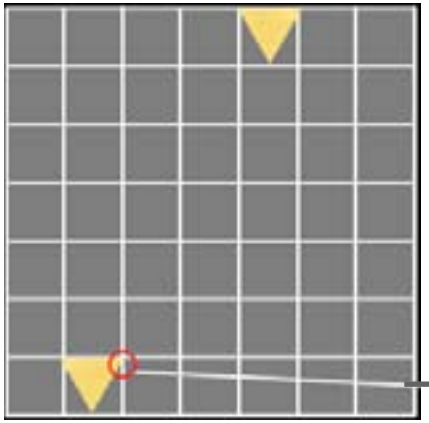


Final shape can be placed randomly anywhere and the orientation is fixed by the system

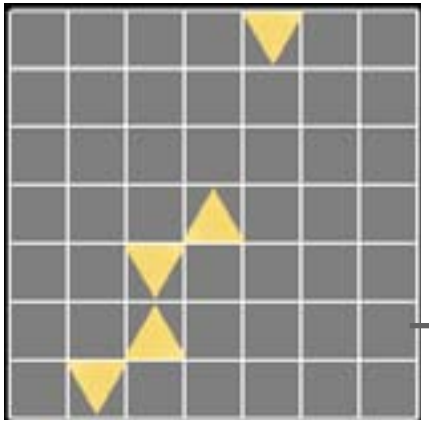
You can place the shape in this Row by pressing 1- 7.



You can select the orientation of the shape.



Select the points to place the flipped shape



Placing the flipped Shape

Figure 6: Shape Game - Prototype

Reference

- M. Tapia, From Shape to Style. Shape Grammars: Issues in Representation and Computation, PhD Thesis, Department of Computer Science, University of Toronto, Toronto, Canada (1996);
- M. Tapia, “A visual implementation of a shape grammar system,” Environment and Planning B: Planning and Design 26 (1999): 59-73.
- G. Stiny, “What is a design?,” Environment and Planning B: Planning and Design 17 (1990): 97-103; G. Stiny,
- G. Stiny, “Introduction to shape and shape grammars,” Environment and Planning B 7 (1980), pp. 349- 351.
- J. Gips, Shape Grammars and their Uses (Birkhauser, Basel, 1975); G. Stiny, Pictorial and Formal Aspects of Shape and Shape Grammars (Birkhauser, Basel, 1975).
- G. Stiny, “Shape: Talking about Seeing and Doing”.

Shape Grammar

The Game

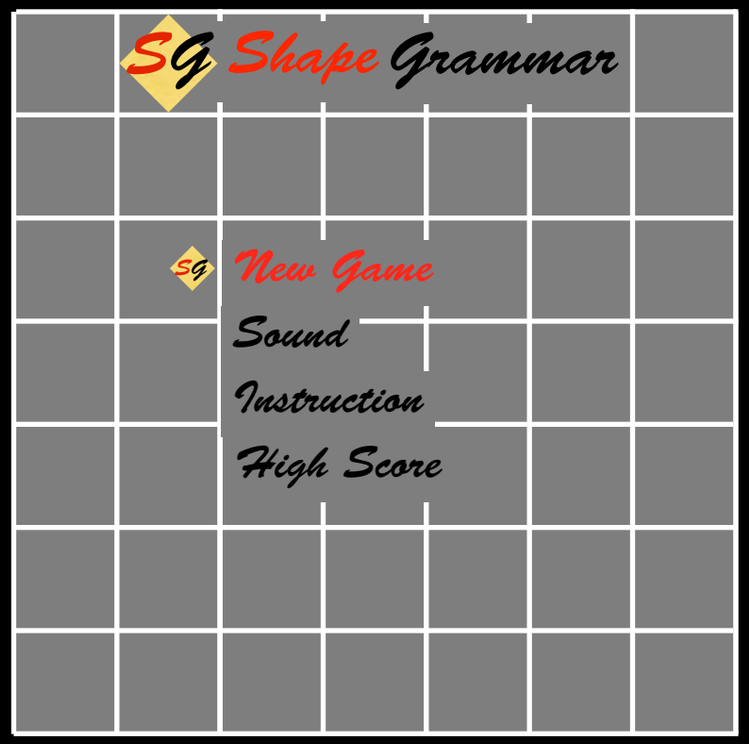
| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |

Scenario Level 1

- The Shape and the rule's are pre defined in each level.
- The final shape placement & orientation is placed by the system.
- The User can select the place of placement of the shape.
- The User can also select the first orientation of the shape.
- The User can select where he wants to place the flipped shape.
- The user has to select the point, and then press ok to put the flipped shape.

| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |

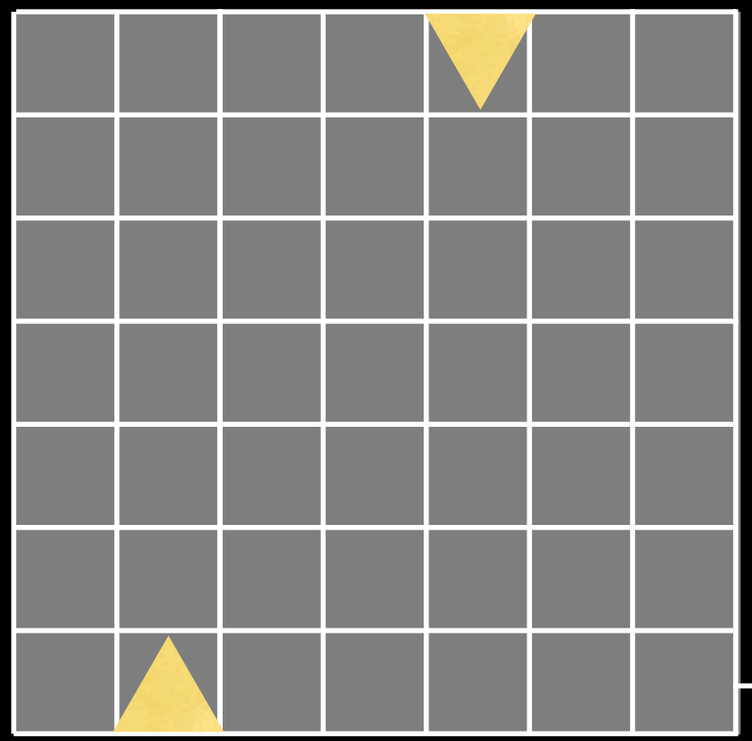
| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |



| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |

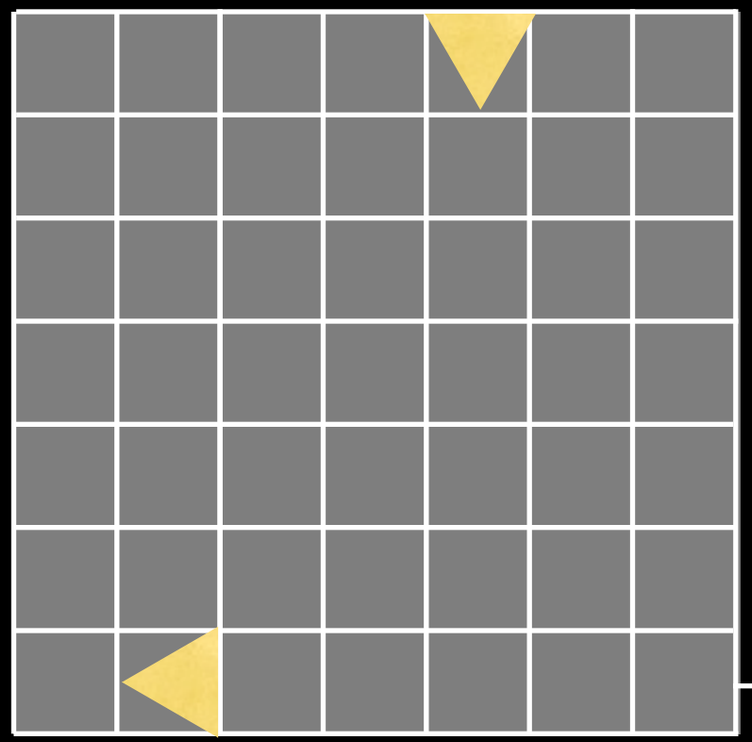
| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |

| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |



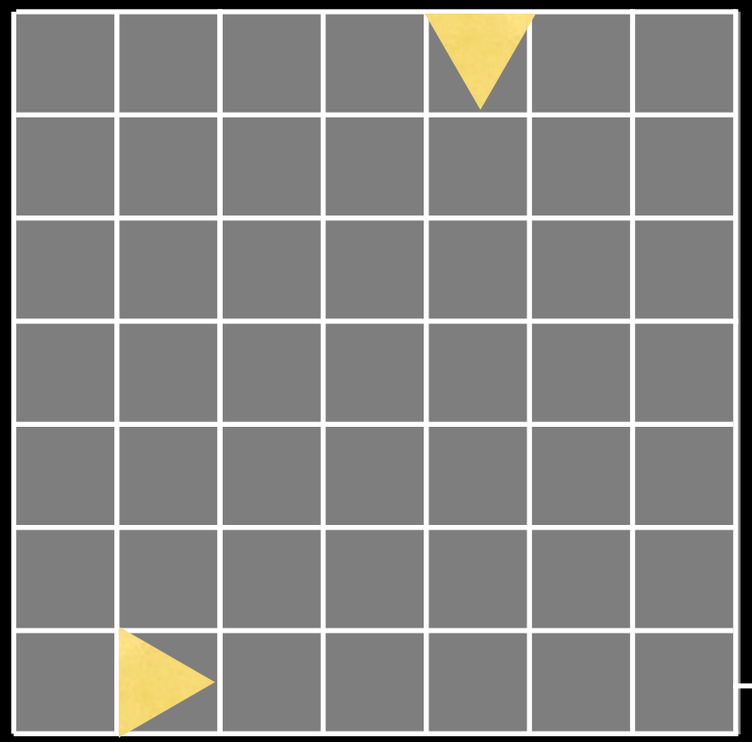
Final shape can be placed randomly anywhere and the orientation is fixed by the system

You can place the shape in this Row by pressing 1- 7.



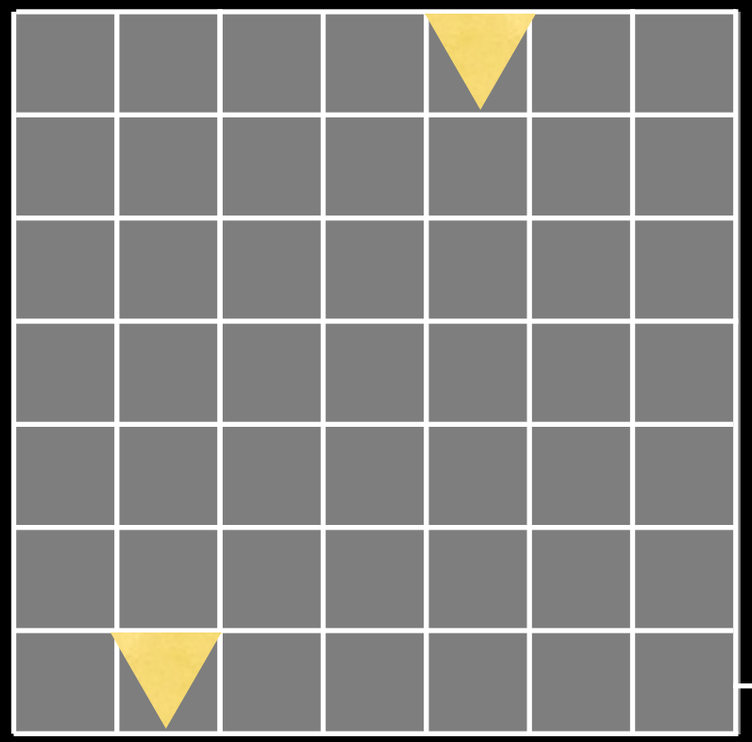
Final shape can be placed randomly anywhere, and the orientation is fixed by the system

You can select the orientation of the shape.



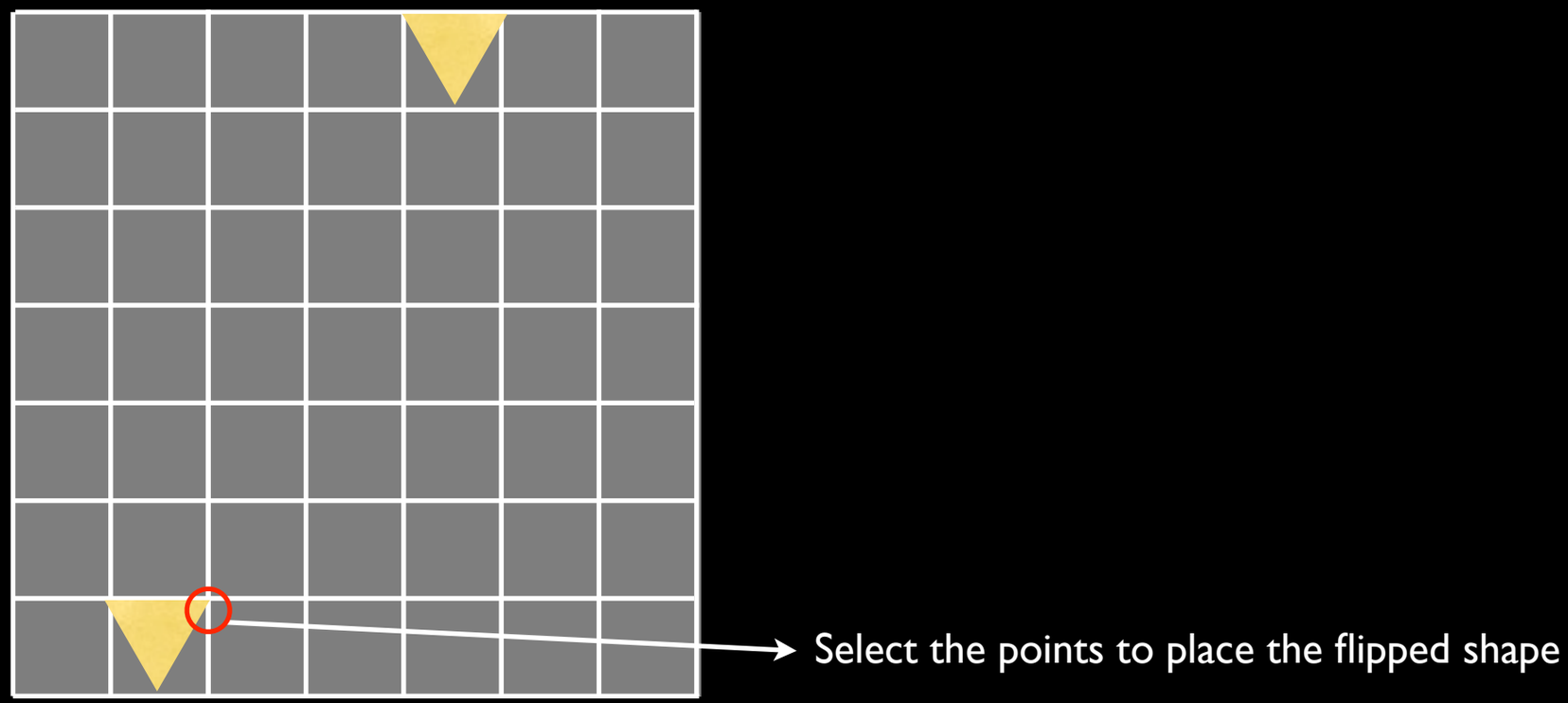
You can select the orientation of the shape.

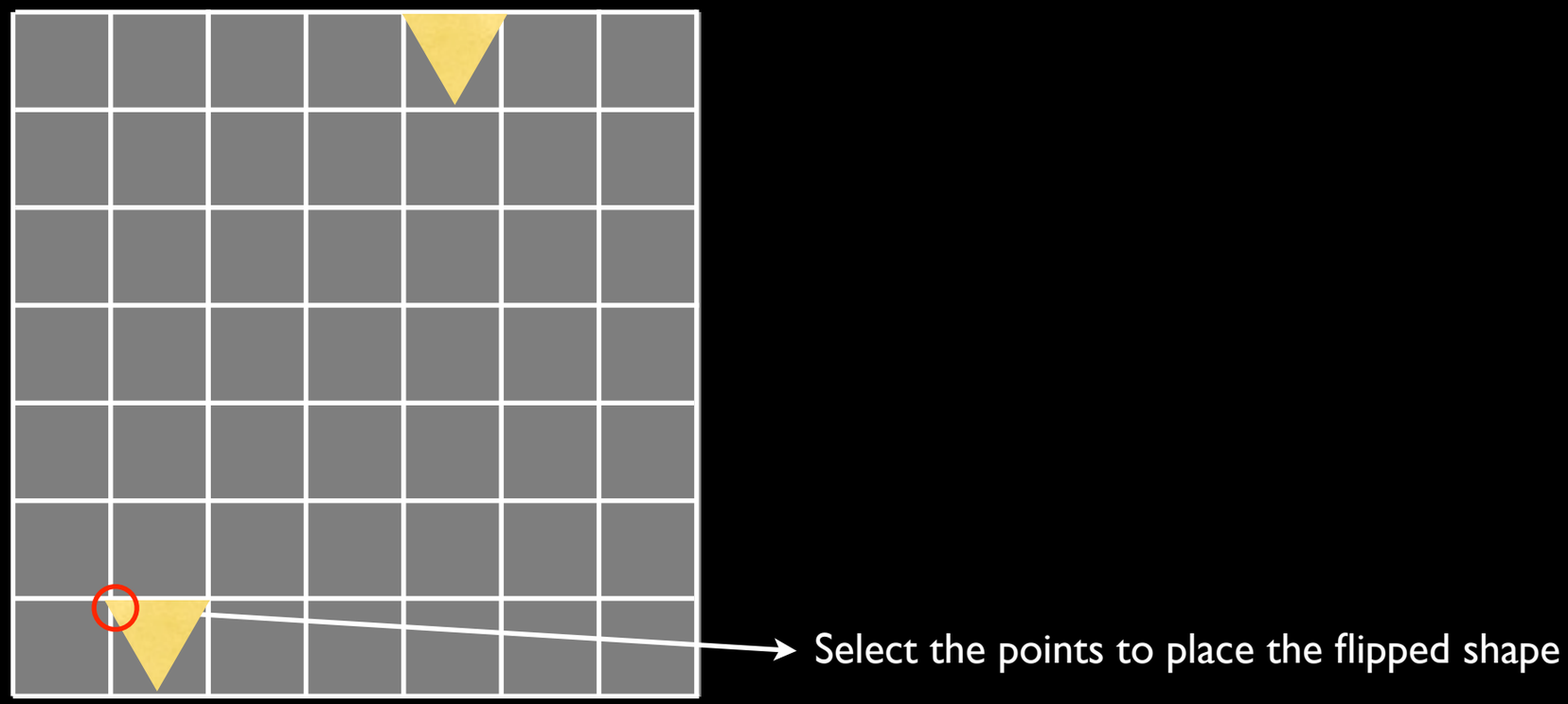
Final shape can be placed randomly anywhere, and the orientation is fixed by the system

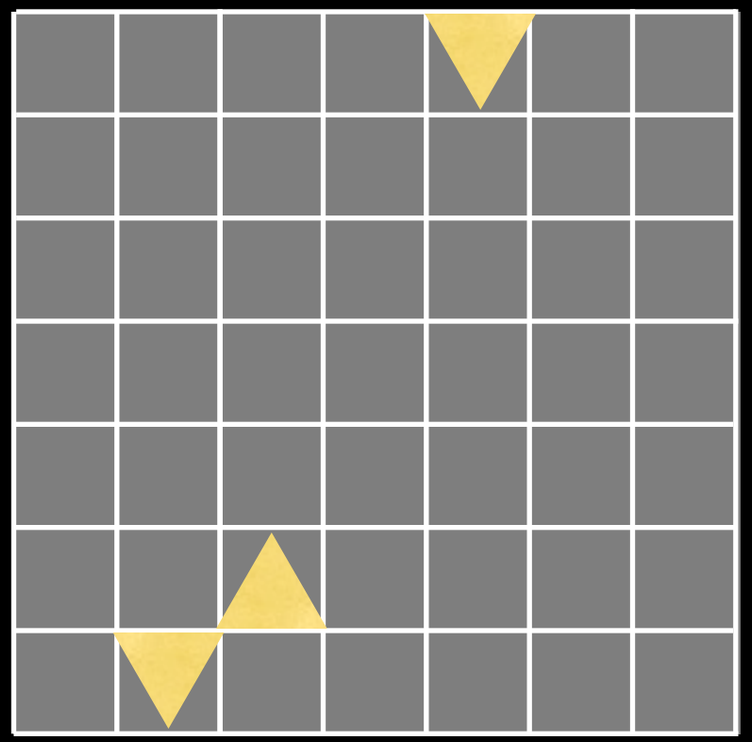


Final shape can be placed randomly anywhere, and the orientation is fixed by the system

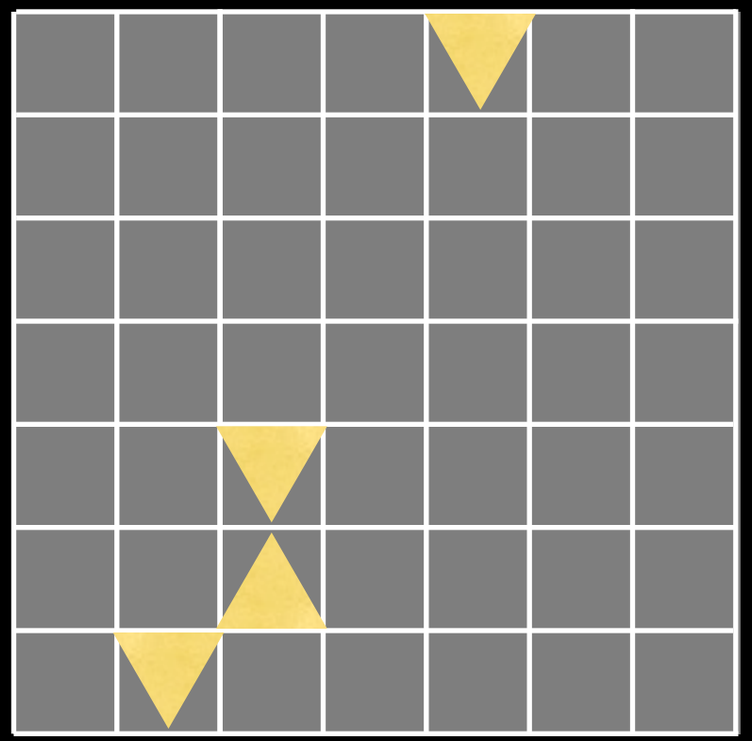
You can select the orientation of the shape.


Select the points to place the flipped shape

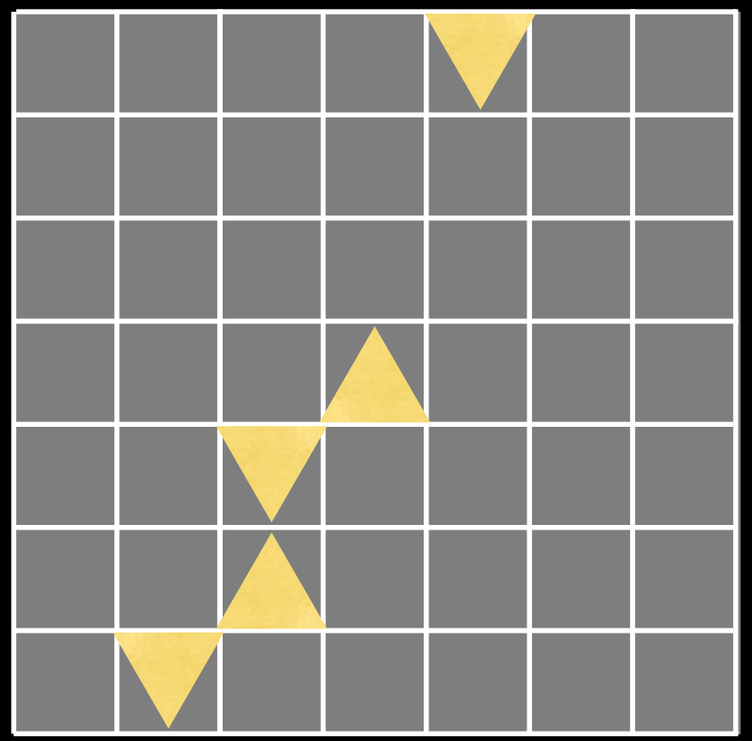

Select the points to place the flipped shape



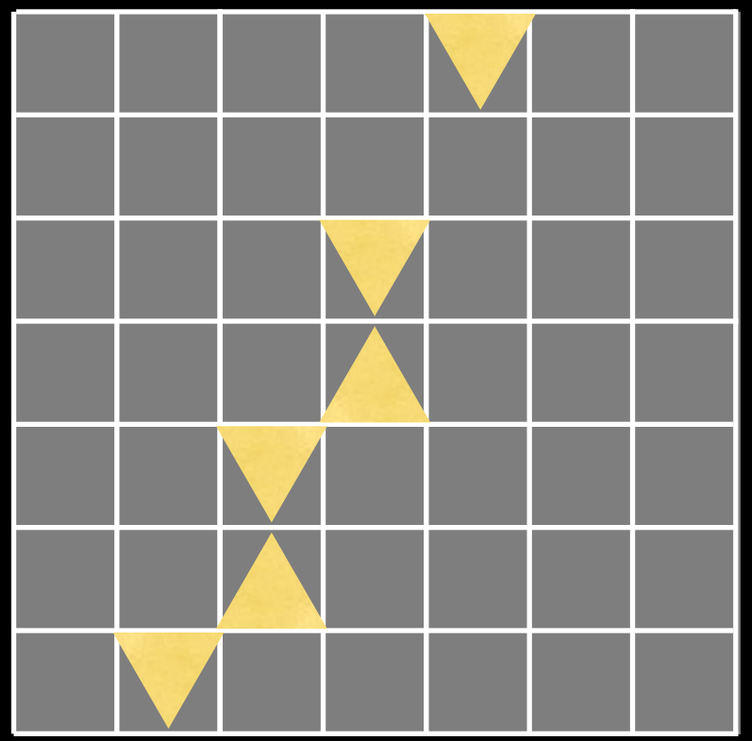
Placed the flipped shape on the selected point



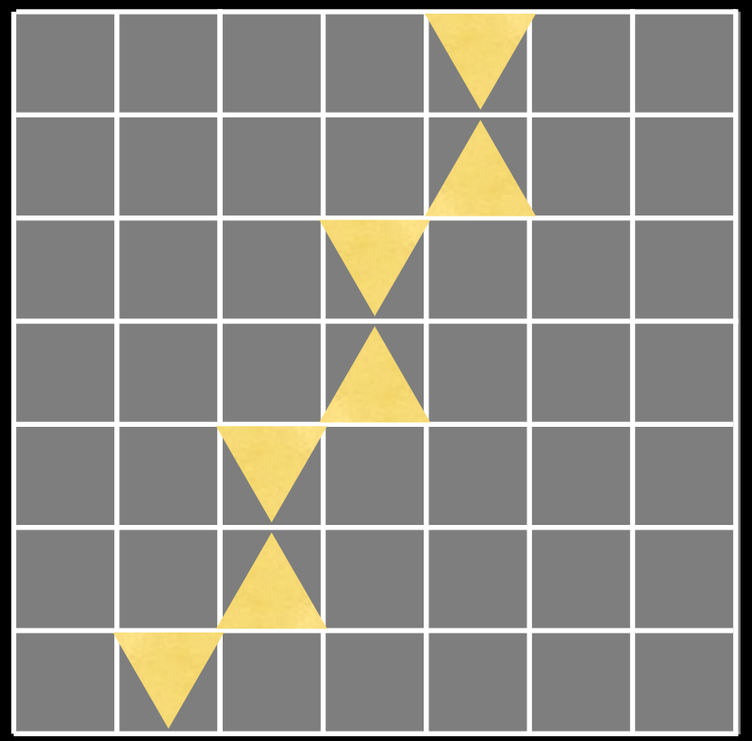
And so on



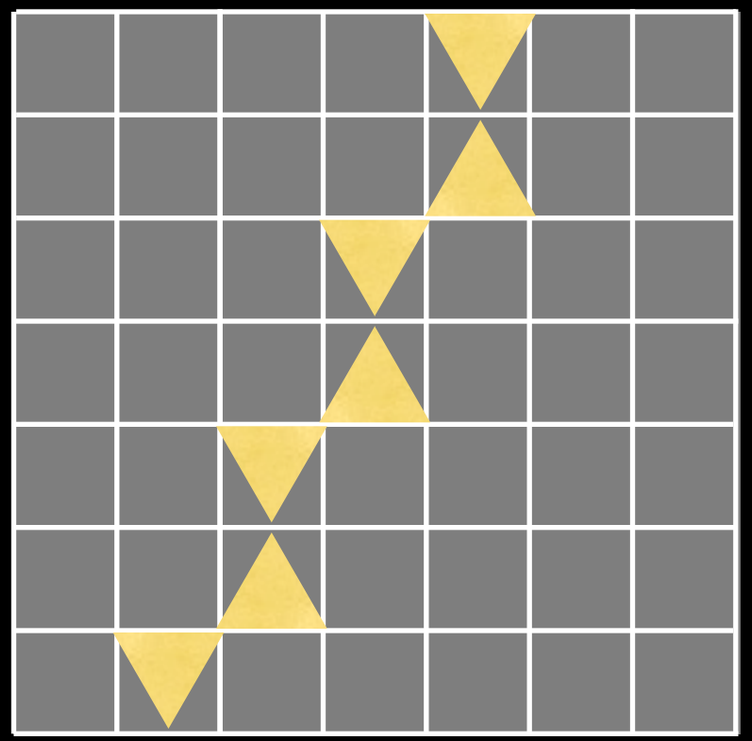
And so on



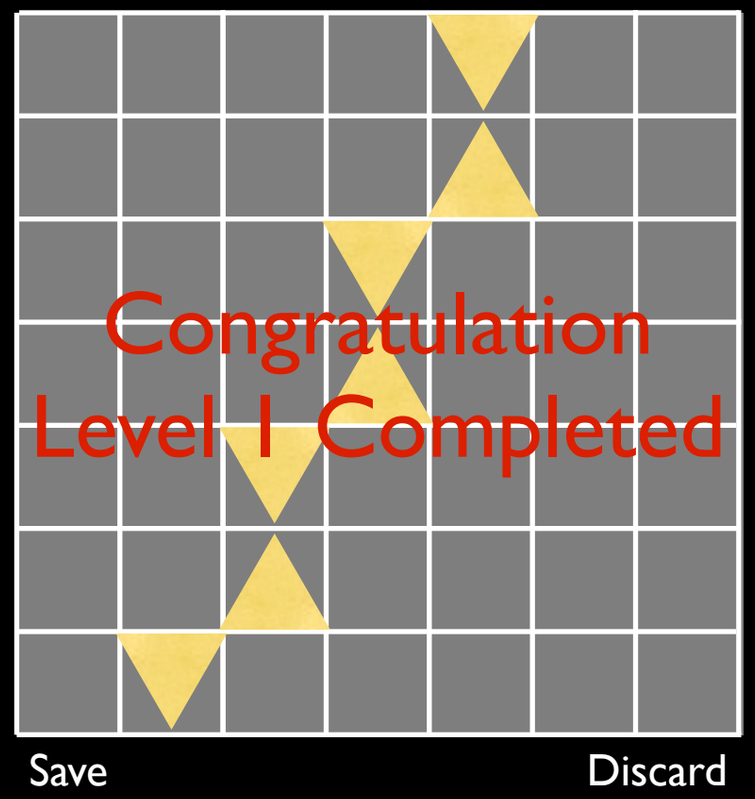
And so on



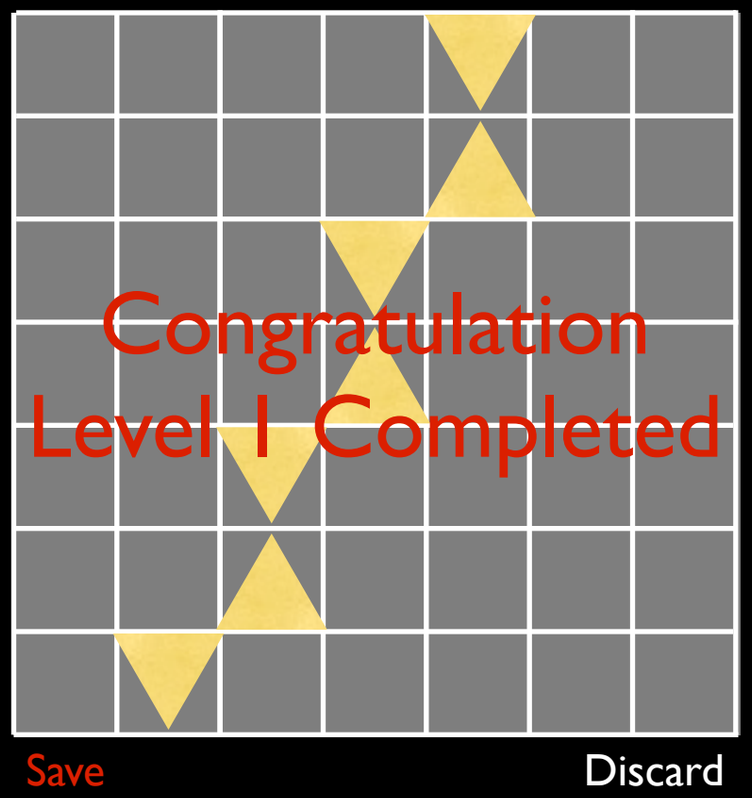
And so on



And Finally we get a Pattern using a shape



You can save or discard the pattern.



You can save or discard the pattern.

Thanks

| Shivendra Singh | DDE | NID | R&D Campus | Bangalore |