DT Challenge Year 7 Blockly – Geometry

Module 1: Drawing Shapes with Turtle

https://groklearning.com/learn/aca-dt-7-bk-geometry/module-1/0/

Previous: none

Next: Angles, Variables and Calculations

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

Computing and geometry can be taught in tandem. Many of the mathematical terms, such as angles and length have a direct representation in computing. By experimenting with these concepts in a computer, students develop a deeper and more intuitive understanding of mathematical concepts whilst at the same time learning about the available computing instruments to produce geometric shapes.

Module outline

Turtle graphics is a term in computer vector graphics, using a relative cursor (the "turtle") upon a Cartesian plane. This module is an introduction to Turtle graphics. It consists of two activities: The first activity lays the foundation of simple move and turn commands that change the position of the turtle on the screen. In the second activity, students learn to read user input as numbers, store it in variables and draw different sized shapes from user input. With user input, students can generate a greater variety of different sized shapes.

Funded by the Australian Government Department of Education and Training.

Guiding Questions

How do we make a computer do exactly what we want?
How can we input and output information with a computer?
How can we store information in a computer (variables)?

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Discussion:
Have you ever seen computer graphics? Maybe in Hollywood movies? How were they made?

Structure of the lesson :

Types of component:

Discussion	**Worksheet**	**Plugged Activity**	
Group Activity	**Unplugged Activity**	**Video**	
Animation	**Reflection**	**Game**	**App**

Lesson Components:

Step 1: Turtle

Plugged Activity

In this activity, students are introduced to the connections between mathematics and computing, teaching geometry through computing. Students learn the first commands, such as move and turn to change the position of the turtle on the screen. By combining these simple commands, students can draw simple geometric shapes, such as a square or a line. In this module, turns are simple 90 degrees left or right turns.

Log on to Grok and start module 1, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-1/0/

Step 2: Variables and Input

Plugged Activity

In this activity, students learn to read user input as numbers, store it into variables and draw different sized shapes from user input. With user input, students can generate a greater variety of different sized shapes. In this activity, students learn that user input changes the behaviour of a program, which results in different output, as represented visually by the turtle graphic.

Log on to Grok and start module 1, problem 7
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-1/7/

Review:

Reflection

Now that you have drawn some shapes, can you think of some other shapes that you would like to draw with Turtle Graphics?

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

None

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 2: Angles, Variables and Calculations
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-2/0/

Previous: Drawing Shapes with Turtle Pen

Next: Parallel Lines and Controlling the

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

Angles and arithmetics are both fundamental to geometry. Combining angles, arithmetics and variables, students can produce a range of different shapes and manipulate existing ones.

Module outline

This module consists of two activities: The first activity introduces the concept of an angle which can be different to the 90 degree angles that students have encountered in the first module. Students are introduced to supplementary angles. With this knowledge students can draw a more complex shape, such as a house. In the second activity students learn about the arithmetic capability of the Blockly programming language and apply this knowledge to convert between external and internal angles, draw lines representing the hands on an analogue clock, and to enlarge geometric shapes.

Guiding Questions

What is an angle?

Funded by the Australian Government Department of Education and Training.

How can shapes be transformed?

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Discussion:
Smartphones usually have cameras. Have you ever re-sized a photo that was taken on a camera? Why did you have to re-size it? How do you think the computer did this?

Structure of the lesson :

Types of component:

Discussion	**Worksheet**	**Plugged Activity**	
Group Activity	**Unplugged Activity**	**Video**	
Animation	**Reflection**	**Game**	**App**

Funded by the Australian Government Department of Education and Training.

Lesson Components:

Step 1: Turtle

Plugged Activity

This activity introduces the concept of an angle which can be different to the 90 degree angles that students have encountered in the first module. Students are introduced to supplementary angles. With this knowledge students can draw a more complex shape, such as a house.

Log on to Grok and start module 2, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-2/0/

Step 2: Variables and Input

Plugged Activity

In this activity, students learn about the arithmetic capability of the Blockly programming language and apply this knowledge to convert between external and internal angles, draw lines representing the hands on an analogue clock, and enlarge geometric shapes. In conjunction with variables (see previous module), students can write simple programs with user input, for example, in order to enlarge an object by a factor that the user specifies.

Log on to Grok and start module 2, problem 6
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-2/6/

Review:

Reflection

Now that you know about computer graphics and some arithmetics, Can you think of a simple algorithm to draw certain shapes? Think about your home, a pet, etc.

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

None

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 3: Parallel Lines and Controlling the Pen

https://groklearning.com/learn/aca-dt-7-bk-geometry/module-3/0/

Previous: Angles, Variables and Calculations

Next: Project: Vector Graphics

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

We often encounter symmetry in the world around us. Think about a parallelogram, or letters, such as H, M, N, or Z. Their symmetry comes from parallel lines and lines that intersect them at particular angles.

Module outline

This module consists of two activities: The first activity introduces the concept of parallel lines with Turtle Graphics. Here, students learn about interrupting the drawing of a line with the aid of the Pen up command. By selectively drawing and moving the pen, dotted and parallel lines can be drawn. In the second activity students learn about the different types of angles that are encountered when another line crosses parallel lines, which is called a traversal. This includes opposite, corresponding, alternate and co-interior angles. With the knowledge about these types of angles, students are led into practical activities, such as drawing the letters Z and N, and drawing a parallelogram with Turtle Graphics.

Funded by the Australian Government Department of Education and Training.

Guiding Questions

How can we draw parallel lines with a computer?
How can we intersect lines to draw shapes and letters?

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Group Activity

Instruct your desk neighbour how to draw a shape that has two opposite parallel lines. You are not allowed to name the shape. Instead, just give your friend instructions as you would do with Turtle graphics. Your neighbour will use a pen and paper to draw the shape. You are not allowed to correct your friend if he/she makes a mistake. When done, compare if the drawn shape matched your instructions.

Discussion

Did the drawn shape match your description? How did you describe the shape? Were the instructions clear enough? How did you describe angles?

Structure of the lesson :

Types of component:

Discussion	**Worksheet**	**Plugged Activity**	
Group Activity	**Unplugged Activity**	**Video**	
Animation	**Reflection**	**Game**	**App**

Funded by the Australian Government Department of Education and Training.

Lesson Components:

Step 1: Parallel Lines: Using the Pen

Plugged Activity

This activity introduces the concept of parallel lines with Turtle Graphics. Students learn about interrupting the drawing of a line with the aid of the Pen up command. By selectively drawing and moving the pen, dotted and parallel lines can be drawn.

Log on to Grok and start module 3, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-3/0/

Step 2: Angles on Parallel Lines

Plugged Activity

In this second activity, students learn about the different types of angles that are encountered when another line crosses parallel lines, which is called a traversal. This includes opposite, corresponding, alternate and co-interior angles. With the knowledge about these types of angles, students are led into practical activities, such as drawing the letters Z and N, and drawing a parallelogram with Turtle Graphics.

Log on to Grok and start module 3, problem 6
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-3/6/

Review:

Reflection

Now that you know more about lines and angles, can you think of other shapes that you could draw with this approach? How would a corresponding Turtle program look like that draws one of these shapes?

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

None

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 4: Project 1: Vector Graphics
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-4/0/

Previous: Parallel Lines and Controlling the Pen | Next: Project: Colours and Loops with Turtle

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

In this module, students consolidate their learnings from the previous modules through practical activities that combine angles, variables, and lines.

Module outline

The module consists of two activities with a total of six practical exercises. In the first activity, students draw the capital letters L and T with Turtle Graphics. In the second activity, students extend this concept to program italic letters L and T and then generalise this approach to let the user choose how much ‘italicisation’ is required and then draw the corresponding letters L or T.

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
Iteraction
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 5: Colours and Loops with Turtle
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-5/0/

Previous: Vector Graphics Next: TBD

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms, Loops
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces involving iteration (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

Graphics looks so much better when it uses colours. In this module, students learn how to set the colours of lines, shapes and backgrounds. They combine this with user input, so that a user can choose a colour at runtime, making programs more interactive.
Looping/iteration is the act of repeating the same or similar steps over and over again until a result has been produced. Looping and automation are strongly connected. In this module, students learn about the foundations of repetition through simple repeat loops. By combining loops with graphics, students can draw shapes, based on their base characteristics alone.

Module outline

This module consists of two activities: The first activity introduces students to setting the colour of the pen, filling shapes, and setting the background colour of the canvas. By combining this with user input, the user can set the colour of objects during the runtime of the program. In the second activity students learn about iteration (loops) and how this concept can be applied to drawing shapes. Rather, than spelling out every single command, students learn to identify a shape's recurring characteristics (such as a line and an angle) and repeat these in a loop to draw the entire

Funded by the Australian Government Department of Education and Training.

shape. By combining this with user input, programs can be developed that draw a great variety of shapes of a given type.

Guiding Questions

How can we colour graphics?
How can we repeat simple commands to produce a more complex shape?

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
Iteration
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Discussion
Imagine a world without colour? How would people get around? Think about traffic lights and smartphone screens. Would a black and white world be easier or harder to live in?

Structure of the lesson :

Types of component:

Discussion **Worksheet** **Plugged Activity**

Group Activity **Unplugged Activity** **Video**

Animation **Reflection** **Game** **App**

Lesson Components:

Step 1: Colours and Loops with Turtle

Plugged Activity

This activity introduces students to setting the colour of the pen, filling shapes, and setting the background colour of the canvas. By combining this with user input, the user can set the colour of objects during the runtime of the program.

Log on to Grok and start module 5, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-5/0/

Step 2: Loops

Plugged Activity

In the second activity students learn about iteration (loops) and how this concept can be applied to drawing shapes. Rather than spelling out every single command, students learn to identify a shape's recurring characteristics (such as a line and an adjacent angle) and repeat these in a loop to draw the entire shape. By combining this with user input, programs can be developed that draw a great variety of shapes of a given type, for example stars that differ by the number of points.

Log on to Grok and start module 5, problem 8
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-5/8/

Review:

Reflection

Now that you know more about colours and loops, can you think of other shapes that you could draw with this approach? How would a corresponding Turtle program look like that draws one of these shapes?

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

None

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 6: More Colours and Loops
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-6/0/

Previous: Colours and Loops with Turtle

Next: Translation and Rotation

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms, Loops
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces involving iteration (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

Colour is fundamental to computer graphics. Any colour is a composite of red, green and blue. By learning about the RGB colour scheme, students can create and manipulate colours in a computer.

Complex shapes can be created by the nested use of loops. By combining nested loops with graphics, students can draw sophisticated shapes.

Module outline

This module consists of two activities: The first activity introduces students to the RED-Green-Blue (RGB) colour representation in which any colour can be represented by its constituting parts of red, green and blue. Students learn about the use of the RGB model to learn about intervals, minima, maxima and their use in Blockly coding blocks.

In the second activity, students go deeper into iteration (loops). They learn how to construct nested loops, that is loops within loops and how they are used to draw repetitive graphics, such as a fence or a chain of daisies.

Funded by the Australian Government Department of Education and Training.

Guiding Questions

How can we represent colour in a computer?
How can we draw complex and repetitive images?

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
Iteration
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Discussion

Our natural world consists of millions of different colours. How can we express them in a way so that we can re-produce them in a computer easily?
How can we construct complex drawings without writing long and repetitive programs?

Structure of the lesson :

Types of component:

Discussion **Worksheet** **Plugged Activity**

Group Activity **Unplugged Activity** **Video**

Animation **Reflection** **Game** **App**

Lesson Components:

Step 1: Colours and Loops with Turtle

Plugged Activity

This activity introduces students to setting the RRB colour scheme. A computer can virtually represent any colour that the human eye can see by specifying the amount of red, green and blue colour components of that colour. Every value of red, green and blue is represented by a number between 0 and 255. By combining 3 numbers, one for red, green and blue, we can make 255x255x255 = 16,581,375 different colours.
In this activity, the RGB concept is then connected to other concepts, such as variables and user input. Students learn that they can ask a user for the colour components, store them in variables and then construct the desired colour.
The problems at the end of this activity combine colours with loops.

Log on to Grok and start module 6, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-6/0/

Step 2: Loops inside Loops

Plugged Activity

In the second activity, students go deeper into iteration (loops). They learn how to construct nested loops, that is, loops within loops and how they are used to draw repetitive graphics, such as a fence or a chain of daisies.

Log on to Grok and start module 6, problem 7
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-6/7/

Review:

Reflection

Now that you know more about making colours and nested loops, can you think of other shapes that you could draw with this approach? For example, could you draw a snowflake or a tree?
For ideas refer to the ‘Tree of Pythagoras’.

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

None

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.

DT Challenge Year 7 Blockly – Geometry

Module 7: Translation and Rotation
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-7/0/

Previous: More Colours and Loops Next: TBD

Key Concepts

Key Concept	Coverage
Abstraction	
Data: collection, representation, interpretation	Data representation as integer and string
Specification, algorithms, implementation	Simple Algorithms, Loops
Digital Systems	
Interaction	Interaction (command line input/output)
Impact	

Objectives (Content Descriptions)

ACTDIK024	Representation / Use of Strings
ACTDIP030	Implement and modify programs with user interfaces involving iteration (disclaimer: visual programming, not text programming used in this course)

What are we learning? (Abstract)

Geometric transformations play an important role in mathematics and in computing. When we move a window on a computer screen, the computer applies a translation algorithm. When we rotate an image on a smartphone, a rotation algorithm ensures that every single point of the image gets rotated correctly.

Module outline

This module focuses on geometric transformations. It consists of two activities: The first activity is about translation, which is a transformation that moves every point in the shape the same distance in the same direction. In the second activity, students learn about rotation, which describes how shapes move around a point in two-dimensional space.

Guiding Questions

What happens inside a computer when we drag a window across the screen with a mouse?
Have you ever had to rotate an image on a smartphone or computer?

Funded by the Australian Government Department of Education and Training.

Elements

Representations
Types of data
Whole numbers represent data
Sequencing
Iteration
User / Environmental Input
Output
Visual programming
Designing (Algorithms)

Purpose/Hook - Computer Graphics

Video

Transformations: https://www.youtube.com/watch?v=1RfEIKQpp3I
The video also contains reflection, which is not part of this module. The corresponding part of the video can be skipped.

Structure of the lesson :

Types of component:

Discussion	**Worksheet**	**Plugged Activity**	
Group Activity	**Unplugged Activity**	**Video**	
Animation	**Reflection**	**Game**	**App**

Lesson Components:

Step 1: Translation

Plugged Activity

In this activity module students investigate translation of geometric shapes. Translation is a transformation that moves every point in the shape the same distance in the same direction. We've already been using translation a lot in our course - quite a few of our loop questions involved some translation. Translation is used in computers all the time to position images and text on the screen. Each image or letter is defined by drawing some lines in a specific direction (just like we explored in the Vector Graphics module), and then translation is used to place it in the correct location on the screen.

Log on to Grok and start module 7, problem 0
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-7/0/

Step 2: Rotation

Plugged Activity

In the second activity, students investigate geometric rotation, which describes how shapes move around a point in two-dimensional space. Students investigate how the choice of the origin (centre of rotation) point changes the result of the rotation operation.

Log on to Grok and start module 7, problem 4
https://groklearning.com/learn/aca-dt-7-bk-geometry/module-7/4/

Review:

Reflection

Other than the examples mentioned already, which interesting shapes could you produce by drawing a simple shape which you then translate or rotate, or both?

Resources and Links (external tools or things teachers need to bring or print)

Printables

None

Digital Resources

Transformations: https://www.youtube.com/watch?v=1RfEIKQpp3I
https://www.mathsisfun.com/geometry/transformations.html
https://en.wikipedia.org/wiki/Rotation_(mathematics)

Feedback Link: ★★★★★

Funded by the Australian Government Department of Education and Training.