

Ticking Time Bomb

Summary

Description

It's a race against time as students help Dash build the Ticking Time Bomb!

Learning Procedure

In the Ticking Time Bomb puzzle students will be introduced to a number of different **Variable** controls that they can edit and program within the course of **9 challenges**. These **Variables** include values in: **Repeat while**, **Repeat until**, **Set**, **Do**, **Change by** and **If blocks**. Students will learn how variables are used to program a course of decisions for the computer to make about which segment of code to run based on events and conditional statements. Students will recognize the ways in which these variables can be used to shortcut a program when a set of instructions repeats within a program. In this case, students are programming Dash to respond to a sequential string of commands to be repeated counting down by 1's from 12.

Concepts Covered

- **Variables** - students will identify and edit variables in the **Set**, **Do**, **Change by**, **If...Else**, **Repeat while** and **Repeat until** blocks.
- **Repeat while**
 - students will locate and identify the **Repeat while** block in the **Variable** controls.
 - students will program a shortcut for a set of repeated instructions to repeat when a condition is true, e.g., **Repeat while** $x > 0$.
- **Set block**
 - students will locate and identify the **Set block** in the **Variable** controls.
 - students will edit the variable, e.g., $x = 12$.
- **Do block**
 - students will locate and identify the **Do block** in **Variable** controls.
 - students will learn to program **Do Eye Lights On** after each **Change by** command.
 - students will recognize other **Do block** variables, including **Look Up/Down**, **Look Left/Right**, **Drive**, **Turn**.
- **Change by**
 - students will locate and identify the **Change by** block in the **Variable** controls.
 - students will learn to edit variables in a **Change by** block by decrements, e.g., **Change x by -1**.
- **If...Else block**
 - students will locate and identify the **If block** in the **Variable** controls.

- students will recognize that the If commands determines which part of the code will be run.
- students will learn how to edit variables in the **IF...Else** block to set conditions that will cue specific segments of the program.
- **Set Wheel Speed** - students will edit left and right wheel directions to program a spin.
- **Sounds** - students will add pre-programmed sounds.

In App

Vocabulary

Variables: a value that can change

Repeat While block: performs a task while in the process of doing something else

Repeat Until block: Blocks inside of the repeat until block will loop until the specific statement is true. If there is any code beneath this block, it will begin.

Do block: lets you choose actions in Dash or Dot's action dropdown

Set block: assigns a value to a variable. This block creates the variable if it does not exist

Set Wheel Speed: a block that allows you to adjust the speed in which the wheels turn

Change by block: changes the value of a variable by an argument. It is essentially an increment or decrement block depending on the sign of the argument.

If...Else: a conditional statement that runs a different set of statements depending on whether the expression is true or false

Reflections Questions

1. What would happen if you changed the **Set** variable from 12 to 6? How would that alter the program?
2. Explain the function of each of the following blocks in the Ticking Time Bomb Puzzle:
Set, Repeat While, Change by.
3. What would happen if you changed the **Change by** block to -2? How about +2?
Experiment with other increments or decrements and see what happens.
4. What would happen if you changed the **Repeat while** block to $x < 8$?
5. Is it possible to set the **Repeat While** block to continually repeat? How is this block different than the **Repeat** or the **Repeat until** blocks?

Extension Activities

**The amount of complexity can be differentiated by adjusting the number and type of conditionals, the level of support and by having students work independently or in a group. *Standards are listed beneath the Solutions.*

1. Ticking Time Bomb II

Challenge students to play around with the Ticking Time Bomb Puzzle. Can they incorporate it into a game? Have students create a game that can be played with 3-4 classmates, using the program from this puzzle, along with some enhancements. What other special effects can they add to make the game exciting and fun? Have students edit command-actions, sounds and variables in existing blocks.

Standard: CC MP 1-8

2. Simon Says Program

Have students draw a program for a game of “Simon Says” incorporating **Set**, **Do**, **Change by**, **If**, **Repeat while** and **Repeat** until blocks. For example, **Set** *the number at which Simon will begin*. What action will players be **Doing**, e.g., hopping on one foot, making a noise, saying something? Set the variable in the **Change by** block, e.g., counting down or up from the Set value. Set two conditionals with two alternative outcomes, e.g., “*If the caller says ‘Simon says do this...’ then the students will follow the instructions of the leader. Else students will not respond.*” Which blocks from the Ticking Time Bomb Puzzle make sense in programming this game? Which do not?

Standards: L.1.6; L.2.6; L.3.6; L.4.6; L.5.6

3. Weather or Not

Have students brainstorm a list of activities they can do outside at home or school when it’s hot, warm, fair, cold or freezing. Choose one weather condition. Example: **If** it’s fair outside today, I can play tag with my friend. Now create a set of **Else** conditionals, e.g., adverse weather, sickness, the grass is being cut, a beehive is hanging from the oak tree in the yard, the ice cream truck arrives, it starts to rain, the temperature rises to over 100 degrees, etc. Have students select an **Else** statement and determine how it will affect their action. Example: Else, there’s an ice cream truck, I will buy an ice cream cone. Have students creatively present their “**If...Else**” statements, either acting them out, drawing them or presenting them in some other creative format, e.g., a song, a poem, a story.

Standards: SL.3.1.D; SL.3.3

4. Write an If...Else Poem

After a unit on poetry and shared reading and discussions of poems. Have students work with a partner to write a 6-line **IF...ELSE Poem**. All of the odd lines should be IF statements. All of the even lines should be Else statements. For example:

IF I win the lottery, I will buy myself a mansion.
Else, I lose my ticket, I will stay in my old home.

IF I grow wings, I will fly high above the school.
Else, someone clips my wings, I will go hurdling down.
IF I grow taller than the highest mountain, I will take care of the world.
Else, I stay as small as a child, my parents will take care of me.

Standards: RL.1.10; RL.2.10; RL.3.10; RL.4.10; RL.5.10

Educational Standards

Common Core Mathematical Practices

- 1. Make sense of problems and persevere in solving them
- 2. Reason abstractly and quantitatively
- 3. Construct viable arguments and critique the reasoning of others
- 6. Attend to precision
- 7. Look for and make use of structure
- 8. Look for and express regularity in repeated reasoning

Next-Gen Science Standards

- 3-5-ETS1-2 Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem

CSTA K-12 Computer Science Standards:

- CL.L1:3-02 Work cooperatively and collaboratively with peers teachers, and others using technology
- CT.L2-01 Use the basic steps in algorithmic problem solving to design solutions
- CT.L2-06 Describe and analyze a sequence of instructions being followed
- CT.L2-07 Represent data in a variety of ways: text, sounds, pictures, numbers
- CT.L2-08 Use visual representations of problem states, structures, and data
- CT.L2-12 Use abstraction to decompose a problem into sub problems
- CT.L2-14 Examine connections between elements of mathematics and computer science including binary numbers, logic, sets, and functions
- CT.L3A-01 Use predefined functions and parameters, classes and methods to divide a complex problem into simpler parts
- CT.L3A-03 Explain how sequence, selection, iteration, and recursion are building blocks of algorithms
- CPP.L1:6-05 Construct a program as a set of step-by-step instructions to be acted out

Solutions

Challenge 1

We've set the orange variable to 12. Press **Start** to see it appear on the upper left corner.

The screenshot shows the Dash Blockly editor interface. On the left is a category menu with options: Start, Drive, Look, Light, Sound, Animations, Control, Variables, and Accessory. The main workspace contains two blocks: an orange 'When Start' block and a blue 'Set' block with a red circle icon and the value '12'. At the bottom, there is a 'START' button on the left, a 'Need a hint?' button in the center, and a 'Reset' button on the right. A progress bar at the very bottom consists of several green segments.

Challenge 2

Press **Start** to see how this program works. The **Do** block lets you use variables in Dash's actions. This block turns Dash's eye light #12 on, and then Dash makes a laser sound.

Start

Drive

Look

Light

Sound

Animations

Control

Variables

Accessory

When Start

Set Eye Light On = 12

Do Eye Light On

Weird Lasers

Reset

Need a hint?

START ▶

◀

▶

Press **Start** to see how this program works! The **Do** block lets you use variables in Dash's actions. This block turns Dash's eye light #12 on, and then Dash makes a laser sound!

Challenge 3

Now rearrange the blocks to make Dash turn the 11th eye light on and then make another laser sound.

The image shows a Scratch code editor with a sequence of blocks:

- When Start
- Set Eye Light On = 12
- Do Eye Light On
- Weird Lasers
- Change Eye Light On by -1
- Do Eye Light On
- Weird Lasers

At the bottom, a challenge box contains the text: "Now rearrange the blocks to make Dash turn the 11th eye light on and then make another laser sound!". The box also features a "START" button, a "Need a hint?" button, and a "Reset" button.

Challenge 4

Now **add 6 blocks** that will turn on the 10th and 9th eye lights after the 12th and the 11th . Make Dash play the laser sound after each light comes on.

When Start

Set [] = 12

Do Eye Light On []

Weird Lasers

Change [] by - 1

Do Eye Light On []

Weird Lasers

Change [] by - 1

Do Eye Light On []

Weird Lasers

Change [] by - 1

Do Eye Light On []

Weird Lasers

Change [] by - 1

Do Eye Light On []

Weird Lasers

Change [] by - 1

Do Eye Light On []

Weird Lasers

Need a hint?

Reset

START ▶

◀

▶

Now add 6 blocks that will turn on the 10th and 9th eye lights after the 12th and 11th. Make Dash play the laser sound after each light comes on.

Challenge 5

That was a lot of work. Let's take a shortcut that will make the rest easier. **Connect the blocks** block to turn lights 12, 11, 10 and 9 on one at a time.

When Start

Set = 12

Repeat while > 8

Do Eye Light On

Weird Lasers

Change by - 1

Need a hint?

Reset

START

That took a lot of work! Let's take a shortcut that will make the rest easier. Connect the blocks to turn lights 12, 11, 10, and 9 on one at a time!

Challenge 6

Edit the **Repeat while** blocks so that the variable **repeat while greater than 0**. This will make all 12 eye lights turn on one by one and make a laser sound starting from 12.

When Start

Set = 12

Repeat while > 0

Do Eye Light On

Weird Lasers

Change by - 1

Need a hint?

Reset

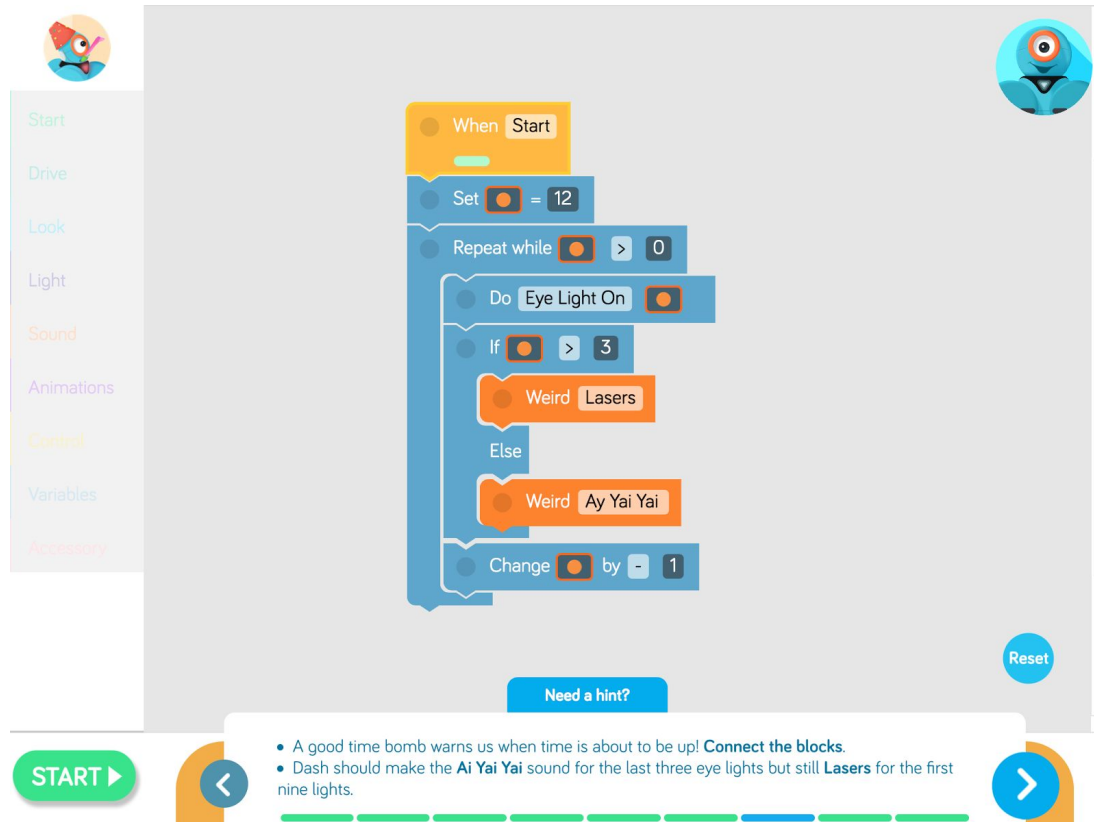
START

Edit the **Repeat while** block so that the variable **repeats while greater than 0**. This will make all 12 eye lights turn on one by one and make a laser sound starting from 12!

Challenge 7

A good time bomb warns us when time is about to be up. **Connect the blocks**

Dash should make the **Ai Yai Yai** sound for the last three eye lights but still **Lasers** for the first nine lights.



The image shows a Scratch code editor interface. On the left is a sidebar with categories: Start, Drive, Look, Light, Sound, Animations, Control, Variables, and Accessory. The main workspace contains the following code blocks:

- When Start
- Set [] = 12
- Repeat while [] > 0
 - Do Eye Light On []
 - If [] > 3
 - Weird Lasers
 - Else
 - Weird Ai Yai Yai
- Change [] by - 1

At the bottom right of the workspace is a "Reset" button. Below the workspace is a "Need a hint?" button. At the very bottom, there is a "START" button, a left arrow, a progress bar with 10 segments (9 green, 1 blue), and a right arrow.

Challenge 8

Now finally make the bomb. After time is up, make Dash start spinning in a circle while sounding the alarm with a **Fire Siren** and flashing the blue and red lights once.

When Start

Set [] = 12

Repeat while [] > 0

Do Eye Light On []

If [] > 3

Weird Lasers

Else

Weird Ay Yai Yai

Change [] by - 1

Set Wheel Speed

Left backward fast

Right forward fast

Reset

Need a hint?

START ▶

◀

▶

Now finally make the bomb! After time is up, make Dash start spinning in a circle while sounding the alarm with a **Fire Siren** and flashing the blue and red lights once.

Weird Lasers

Else

Weird Ay Yai Yai

Change [] by - 1

Set Wheel Speed

Left backward fast

Right forward fast

Transport Fire Siren

All Lights [blue]

All Lights [red]

Reset

Need a hint?

START ▶

◀

▶

Now finally make the bomb! After time is up, make Dash start spinning in a circle while sounding the alarm with a **Fire Siren** and flashing the blue and red lights once.

Challenge 9

Now **add a block** that makes Dash **repeat** the spinning, siren, and flashing blue and red lights **until** you press Dash's top button. Test it by pressing the **top button** immediately after the **red light**.

The image shows a Scratch-like block editor interface for Dash the Robot. The interface includes a sidebar on the left with categories: Start, Drive, Look, Light, Sound, Animations, Control, Variables, and Accessory. The main workspace contains a script with the following blocks:

- Orange block: Weird Lasers
- Else block
- Orange block: Weird Ay Yai Yai
- Change block: Change [] by - 1
- Repeat until block: Repeat until Dash Top Button
- Set Wheel Speed block: Left backward fast, Right forward fast
- Transport block: Transport Fire Siren
- All Lights block: All Lights [blue]
- All Lights block: All Lights [red]

At the bottom of the workspace, there is a "Need a hint?" button and a "Reset" button. Below the workspace, there is a "START" button and a progress bar with 10 segments, the 9th of which is highlighted in blue. A hint box is open, displaying the following text:

Now **add a block** that makes Dash **repeat** the spinning, siren, and flashing blue and red lights **until** you press Dash's **top button**. Test it by pressing the **top button** immediately after the **red light**