

# code jumper



## **Lesson 11**

Selection and Random

# Code Jumper Curriculum: Lessons



## Lesson 11 Selection and Random

American Printing House for the Blind

*Code Jumper Curriculum: Lessons*

Copyright © 2020 American Printing House for the Blind

All rights reserved. No part of this publication may be reproduced, stored in retrieval system, or transmitted in any form or by any means, except as expressly permitted under copyright law, without written permission of the publisher.

Published by American Printing House for the Blind

1839 Frankfort Avenue, Louisville, KY 40206

[www.aph.org](http://www.aph.org) | [info@aph.org](mailto:info@aph.org)

# OVERVIEW

## LESSON OBJECTIVES

Students will:

- Be able to develop programs that use selection and random values

## EXPECTED OUTCOMES

- All students: Be able to describe what selection with random values is
- Most students: Be able to recreate programs that use selection and random values
- Some students: Be able to create original programs that use selection, random values, and loops

## LESSON PLAN STRUCTURE

- Unplugged Activity
- Guided Code Jumper Activity
- Exploration
- Standards and Check for Understanding

## RESOURCES

- Code Jumper Tutorial Videos
  - Code Jumper App: <https://www.youtube.com/watch?v=vg72YPz6CWY>
  - The Hub: <https://www.youtube.com/watch?v=KGb51PW9zJQ&lis=>
  - Play and Pause Pod: <https://www.youtube.com/watch?v=446jCw8qcDI&t=>
  - Selection and Merge Pod: <https://www.youtube.com/watch?v=Lx7EnNcjUrU>
  - Plugs: <https://www.youtube.com/watch?v=hTy8moSohIE>

- Loop Pod: <https://www.youtube.com/watch?v=EFKbKLID3HI>
- Code Cards

## KEY VOCABULARY

- **Selection:** A structure in computer programming where if a question is asked, the program decides what to do next based upon the answer (This is sometimes referred to as an if-then-else statement.)
- **Random Value:** A value that changes in a computer program depending upon the possible outcomes of the programming sequence (Compare to: Rolling dice)

# UNPLUGGED ACTIVITY

## OBJECTIVE

Introduce the idea of a selection structure and random values in computer programming and how students will apply it to their own program.

## VOCABULARY

- **Random Value:** is a value chosen by chance from some specified group of values. The value is not known until it has been generated.

## MATERIALS

A set of five tactile dice per student (Put dice in small, snack-sized bags for easy distribution.)

Random Value Counting Page (An example is at the end of the lesson.)

## INSTRUCTIONS

1. Divide students into groups of three to four. Distribute the Random Value Counting Page to each player. It can also be distributed electronically or in the Computer Science Journals.

2. Explain to students that they will be playing a dice game with their small group and will keep track of the random values of their dice on their Random Value Counting Page.
3. Instruct students to listen to the directions. The teacher may also want to model how to play.
  - a. Directions for playing "Random Values"
    - i. For Game #1: One player in the group calls out a number between one and six.
    - ii. Each player rolls all of the dice from their bag. This counts as roll number one.
    - iii. Any dice that match the number chosen can be set aside.
    - iv. Players roll all of the dice that did not match the chosen number and count that as roll number two.
    - v. Players repeat this process until all dice are showing the chosen number.
    - vi. Players keep track of the number of rolls it takes them on their Random Value Counting Page.
    - vii. When all players have finished round one, a different player calls out a number between one and six and the group repeats this process.
  - For example, if the student chooses the number 5, then they keep rolling all the dice trying to get each one to be the number 5. Therefore, the number of dice that are being rolled will decrease until finally they have all dice showing the same number. Students will record the number they are trying to get.

Game options: The teacher can decide if there is a winner each time. The student with the fewest number of rolls is declared the winner.

4. Once all groups have completed their games and recorded their results, have them work together to come up with three observations about the data they collected on the Random Value Counting Page.
5. Instruct students to write three of their group observations in their Computer Science Journals.
6. Ask each small group to choose one group member to share their observations with the class. As groups share, the teacher will collect the observations.

## CLOSURE

Review with students the definition of random value. Help them make the connection that in the dice game the random value was the number of rolls it took to get all the dice to show the same number. Ask students to make inferences about random values in our world.

# CODE JUMPER GUIDED ACTIVITY

## OBJECTIVE

Students will be able to develop programs that use selection and random values.

## VOCABULARY

**Random Value:** is a value chosen by chance from some specified group of values. The value is not known until it has been generated.

## MATERIALS

- Code Jumper Kit
- Play pods

- Selection pod
- Merge pod
- Random plug

## INSTRUCTION

1. Instruct students to take 4 Play pods, 1 Selection pod, 1 Merge pod, and the Random plug out of the kit.
2. Review the Selection pod and the Merge pod. Explain that we use selection to decide which path a program will take, like coming to a fork in the road and deciding to go one way or the other. The Merge pod brings the program back together, like two paths becoming one.
3. Ask students to connect the Selection pod to a thread on the Hub. Next, add one Play pod to each of the ports adjacent to the spokes on the Selection pod. Students can select whatever Sound Set they would like, making sure that each sound is different.
4. Explain that both dials on the Selection pod can be set to any value between 1 and 8, and that if the statement is true, the program will choose the path of the two-spoke dial. If the statement is false, the program will choose the path of the three-spoke dial.
5. Ask students to set the value for each dial on the Selection pod and predict which path the program will take. Repeat this task several times until the students understand the concept.
6. Introduce the Random plug. Ask students to remove the Random plug from the Code Jumper Kit. The Random plug is similar to the Constant plug, but instead of having a fixed value, the Random plug generates a random value between 1 and 8 each time the program runs. Explain that a random value is a number that is randomly generated by a computer. This is similar to rolling dice, as you do not know what value you will get.

7. Give students time to create a program using the Selection pod. The students should create a program that contains the Hub, 4 Play pods, the Selection pod, the Merge pod, and (optionally) the Pause pod. Students can use a Sound Set of their choosing. Remind students to debug their programs.
8. Ask students to record their code in their Computer Science Journals. Draw their attention to how the statement is read when the Selection pod is connected. Ask students why END IF is important (Expected response: IF is the end of the conditional statement).
9. Have students insert the Random plug into the two-spoke dial on the Selection pod without pressing Play. Draw their attention to how the statement is read when the Random plug is inserted (Expected response: the word random will appear in the first line of the IF statement).
10. Then, ask students to set the three-spoke dial to 3 and predict what might happen when they press Play. Ask students to write their prediction in their Computer Science Journals. (Expected response: If the statement is true, the program will choose the path of the two-spoke dial. If the statement is false, the program will choose the path of the three-spoke dial.)
11. Ask students to run the program several more times so they see that the value on the two-spoke dial changes when they use the Random plug.

## CLOSURE

- Ask students to explain what selection is and to give an example of how they used it in this lesson.
- Ask students what the Random plug does and to give an example of how they used it in this lesson.



# EXPLORATION

## OBJECTIVE

Students will create a story program with multiple endings.

## VOCABULARY

- **Random Value:** is a value chosen by chance from some specified group of values. The value is not known until it has been generated.

## MATERIALS:

- Code Jumper Kit: Hub, Play pods, Loop pods, Pause pods, Selection pod, Merge pod
- Random and Selection Code Card 1: Helen's Story with Loop
- Random and Selection Code Card 2: Helen's Story with Loop and Random Plug
- Helen's Story Card 1
- Helen's Story Card 2

## INSTRUCTION

1. Give students a copy of Selection and Random **Code Card 1:** Helen's Story With a loop.
2. Ask students to create the program for Helen's Story using Code Card 1: Helen's Story with a loop. Prompt students to think about how to bring a program back together after the Selection pod in a program. (Expected response: Use the Merge pod.) **Tip:** A way to describe the Selection and Merge pods is to say it's like coming to a fork in the road, choosing one path, and then going back to the main road.
3. Ask students to play the program and see which path the Selection pod took. Ask students: What would you need to change to take the other path of the Selection pod? (Expected response: Change the dials on the Selection pod.)

Give the students a few minutes to run the programs while changing the dials.

4. Ask students: How do you make the program repeat? (Expected response: use a loop.)
5. Ask students to add the program they made inside a loop and set the Loop pod to loop four times.
6. Have students find the Random plug in the kit. Remind students that the Random plug generates a random value between 1 and 8 each time the program runs. In this program it is being used in the Selection pod, but it can also be used other Commands pods as well.



[Figure Caption:] The Random plug, with raised lowercase letter r

7. Insert the Random plug into the two-spoke dial on the Selection pod and set the three-spoke dial to 4.
8. In their Computer Science Journals, have students record how the story ends for the four times that it is played in the loop.

a. Example:

Loop Number	Ending: Two-spoke = 1 Three-spoke = 2
1	
2	
3	
4	
Second run-through	
1	
2	
3	
4	

9. Change the value on the three-spoke dial and see if that changes the result and try a few more runs. Record in the Computer Science Journal.
10. Distribute Selection and Random Code Card 2: Helen's Story With a loop and Random plug. Students will insert the Random plug into the two-spoke dial of the Selection pod and set the three-spoke dial to 4. Tip: setting the three-spoke dial to 4 increases the chances of hearing both stories.

## CLOSURE

- Ask students: Did you notice a difference when the three-spoke dial was changed from 4? Did you hear an ending more frequently? Why do you think that is?
- Ask students: Why would using the Random plug be useful when creating a program?

# STANDARDS AND CHECK FOR UNDERSTANDING

## CSTA STANDARDS\*

- 1A-AP-12: Develop plans that describe a program's sequence of events, goals, and expected outcomes.
- 1B-AP-10: Create programs that include sequences, events, loops, and conditionals.

## NATIONAL CURRICULUM OF ENGLAND\*\*:

### Key Stage 1:

- Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
- Create and debug simple programs
- Use logical reasoning to predict the behavior of simple programs
- Use technology purposefully to create, organize, store, manipulate and retrieve digital content
- Recognize common uses of information technology beyond school
- Use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies

### Key Stage 2:

- Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output

- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.
- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.
- Use technology safely, respectfully and responsibly; recognize acceptable/unacceptable behavior; identify a range of ways to report concerns about content and contact.

## CLOSING ACTIVITIES AND CHECK FOR UNDERSTANDING

Ask students to write a scenario in their Computer Science Journals where a random selection is made (Example: Pulling a certain type of candy out of a bag)

Check for Understanding	Completed
Students can describe what happens when a selection is made that is random.	Yes / No

\*Computer Science Teachers Association (2017). CSTA K-12 Computer Science Standards, Revised 2017. Retrieved from <http://www.csteachers.org/standards>

\*\*Education, Department for. "National Curriculum in England: Computing Programmes of Study." GOV.UK, 11 Sept. 2013, [www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study](http://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study)

## CODE CARDS FOR SELECTION AND RANDOM LESSON

Student Selection and Random **Code Card 1**: Helen's Story With a Loop

- Sound category: Sample Sounds
- Sound Set: Helen's Story

THREAD 2 Helen's Story

LOOP 4 times

PLAY Helen is looking for her friends for 1 times speed

PLAY She thinks they might be in the for 1 times speed

PLAY Haunted house for 1 times speed

PLAY She opens the door for 1 times speed

IF 5>8

PLAY Her friends jump out and yell surprise! for 1 times speed

ELSE

PLAY She sees a ghost and screams! for 1 times speed

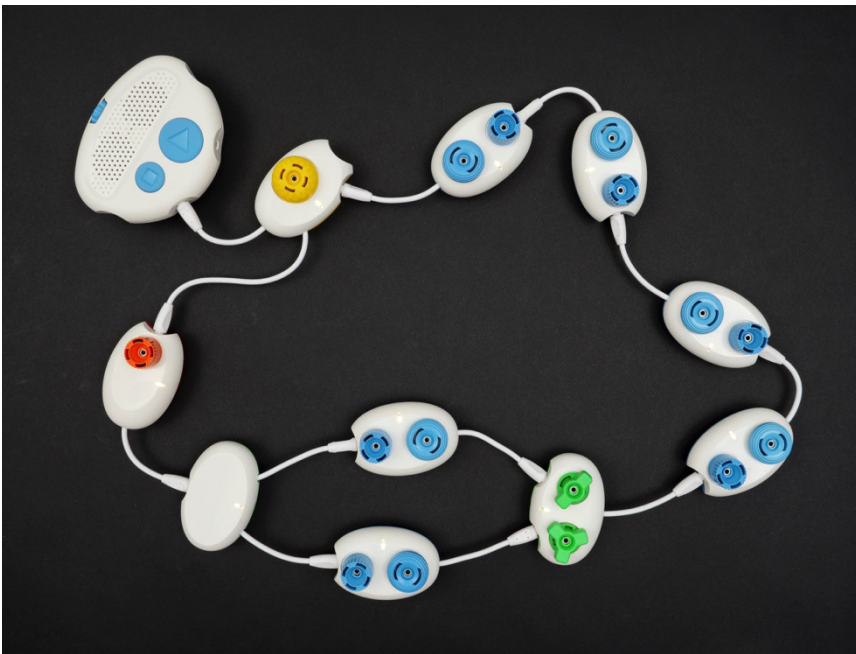
END IF

PAUSE for 1 beat

END LOOP

END THREAD

```
Thread 2 Helen's Story
LOOP 4 times
  PLAY Helen is looking for her friends for 1 times speed
  PLAY She thinks they might be in the for 1 times speed
  PLAY Haunted house for 1 times speed
  PLAY She opens the door for 1 times speed
  IF 5 is greater than 8
    PLAY Her friends jump out and yell surprise! for 1 times speed
  ELSE
    PLAY She sees a ghost and screams for 1 times speed
  END IF
  PAUSE for 1 beat
END LOOP
End Thread
```



[Figure caption:] The screenshot at the top shows a Code Jumper program in Thread 2. Under Thread 2 Helen's Story, the

commands read, LOOP 4 times, PLAY Helen is looking for her friends for 1 times speed, PLAY She thinks they might be in the for 1 times speed, PLAY Haunted house for 1 times speed, PLAY She opens the door for 1 times speed, IF 5 is greater than 8, PLAY Her friends jump out and yell surprise! for 1 times speed, ELSE, PLAY She sees a ghost and screams for 1 times speed, END IF, PAUSE for 1 beat, END LOOP. The commands are followed by End Thread. The photo below the screenshot displays a Code Jumper program that incorporates the Loop, Selection, Merge, Pause, and Play pods. The pods are arranged as follows: The Loop pod is connected to the Hub at Port 2 by its shorter wire. Four Play pods are connected, one after the other, to the Loop pod's lower-right port. The Selection pod is connected to the last Play pod in that sequence. Then, one Play pod is connected to each of the Selection pod's ports, and each of the Merge pod's two wires is connected to one of these two Play pods' ports. The last pod in the looped program is the Pause pod, and it is connected to the Loop pod's longer wire.



## Student Selection and Random **Code Card 2**: Helen's Story With Loop and Random plug

- Sound category: Sample Sounds
- Sound Set: Helen's Story

THREAD 1 Helen's Story

LOOP 4 times

PLAY Helen is looking for her friends for 1 times speed

PLAY She thinks they might be in the for 1 times speed

PLAY Haunted house for 1 times speed

PLAY She opens the door for 1 times speed

IF random > 4 then

PLAY Her friends jump out and yell surprise! for 1 times speed

ELSE

PLAY She sees a ghost and screams! for 1 times speed

END IF

PAUSE for 1 beat

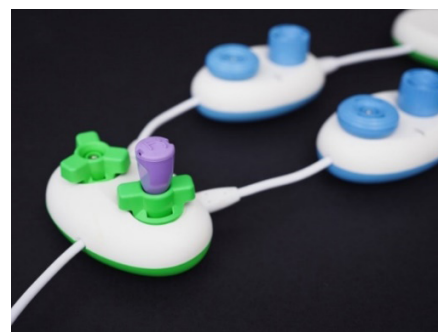
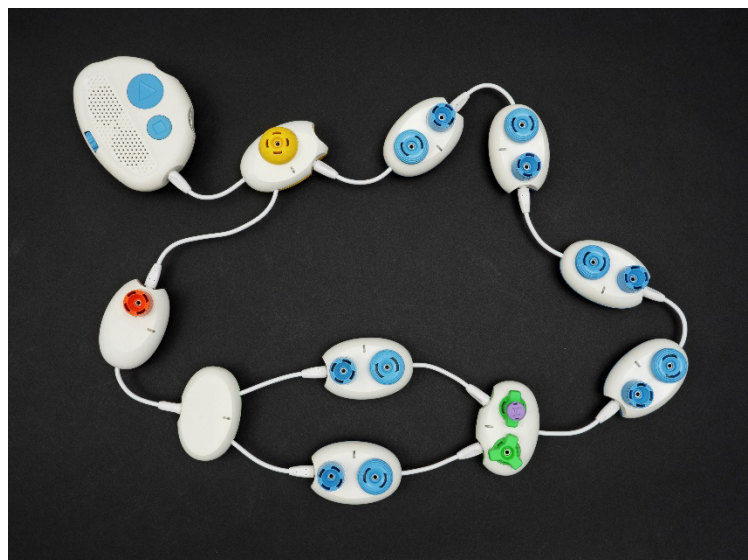
END LOOP

END THREAD

```

THREAD 1 Helen's Story
  LOOP 4 times
    PLAY Helen is looking for her friends for 1 times speed
    PLAY She thinks they might be in the for 1 times speed
    PLAY Haunted house for 1 times speed
    PLAY She opens the door for 1 times speed
    IF random is greater than 4
      PLAY Her friends jump out and yell surprise for 1 times speed
    ELSE
      PLAY She sees a ghost and screams for 1 times speed
    END IF
    PAUSE for 1 beat
  END LOOP
END THREAD

```

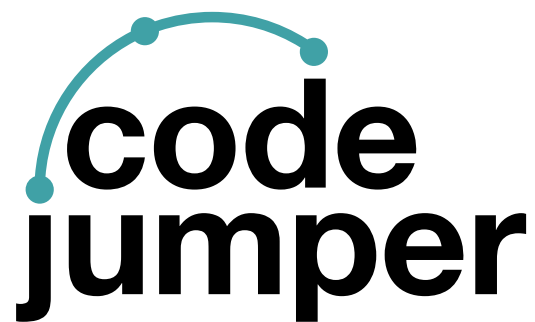


[Figure Caption:] At the top is a screenshot of the Code Jumper app with a program in Thread 1. Under THREAD 1 Helen's Story, the commands read, LOOP 4 times, PLAY Helen is looking for her friends for 1 times speed, PLAY She thinks they might be in the for 1 times speed, PLAY Haunted house for 1 times speed, PLAY She opens the door for 1 times speed, IF random is greater than 4, PLAY Her friends jump out and yell surprise for 1 times speed, ELSE PLAYShe sees a ghost and screams for 1 times speed, END IF, PAUSE for 1 beat, END LOOP; the commands

are followed by END THREAD. Below this screenshot are two photos. The photo on the left displays a Code Jumper program containing a Loop pod, several other pods, and a Random plug. One end of the Loop pod is connected directly to the Hub at Port 1. The remaining pods are connected, one after the other, to the port on the lower-right side of the Loop pod. The order of the pods within the Loop is as follows: Play pod, Play pod, Play pod, Play pod, Selection pod with one Play pod connected to the port adjacent to the two-spoke dial and a second connected to the port adjacent to the three-spoke dial, Merge pod, Pause pod. The Counter plug is connected to the port in the Selection pod's two-spoke dial. The photo on the right, taken from a side angle, focuses on the selection portion of the program.

## RANDOM VALUE COUNTING PAGE PLAYER CARD

	Player #1: Number of Rolls	Player #2: Number of Rolls	Player #3: Number of Rolls	Player #4: Number of Rolls
Game #1 Value:				
Game #2 Value:				
Game #3 Value:				
Game #4 Value:				



For more resources, visit [codejumper.com](http://codejumper.com)

Copyright © 2020



**AMERICAN  
PRINTING  
HOUSE**

**1839 Frankfort Avenue  
Louisville, Kentucky 40206  
502-895-2405 • 800-223-1839  
[www.aph.org](http://www.aph.org) • [info@aph.org](mailto:info@aph.org)**