

Brownie Coding for Good

Find out how to create apps and games that help others by earning these three badges!

Badge 1:
Coding Basics

Badge 2:
Digital Game Design

Badge 3:
App Development



This booklet gives girls an overview of the badge requirements and badge steps for all three Brownie Coding for Good badges. It also includes interesting background information to spark girls' interest in coding. Volunteers can access the Volunteer Toolkit (VTK) to find complete meeting plans, including detailed activity instructions and handouts.

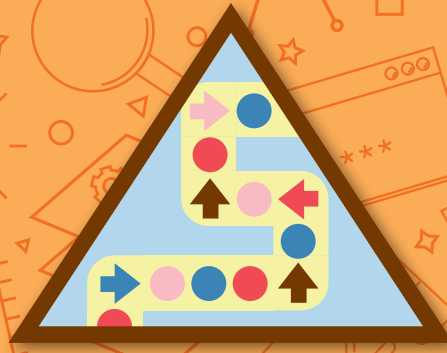
Welcome to the world of coding.

When you've earned these three badges, you'll know how people create code that tells computers what to do.

- You'll know how to write a computer program.
- You'll know how video games are created.
- And you'll know how people make apps.

You'll also learn about important female computer scientists and how to use computers to solve problems and help people.

Volunteers can access the Volunteer Toolkit (VTK) to find complete meeting plans, including detailed activity instructions and handouts.



Badge 1: Coding Basics

Laptops, tablets, and smartphones are all computers. We use computers every day, and they help us in lots of ways.

What makes all those computers work? People write instructions telling the computer what to do. By exploring how computer scientists write programs, you'll learn how computers work, too!

Steps

1. Create algorithms for a computer that follow a sequence
2. Use loops to improve your algorithm
3. Use events to make things happen
4. Learn about women in computer science
5. Create your own set of commands that use events

Purpose

When I've earned this badge, I'll know how programmers write programs that make computers work and how people can use computers to help others.

What's a Programmer?

A long time ago, people—mostly women—were the first “computers,” doing complicated math problems.

Today a **computer** is a machine that can follow directions. A laptop is a computer, but so is a smart watch.

For a computer to work, it needs to follow instructions that have been written in a **code** it understands.

Programming is when people code a sequence of instructions, or an **algorithm** for a computer.

Computer scientists are sometimes called programmers or coders.

STEP

1 Create algorithms for a computer that follow a sequence

How is writing a computer program like making a cake? When you make a cake, you follow the recipe carefully.

You also do each step of the recipe in a certain order. For example, you have to grease the cake pan before you put in the batter. If you don't, the cake will stick.

Writing a computer program is like making a cake. You write the directions for the computer to follow. The directions are called an **algorithm**.

When you write a computer program, you also make sure the directions, or steps, are in the right order. That's called a **sequence**.



WORDS TO KNOW

Algorithm This is a set of step-by-step instructions for how to do something. A recipe is an algorithm. It tells you all the steps you need to take to cook something. When a friend gives you directions to her house, that's an algorithm, too. She's telling you the steps you need to take to get to her house.



Debugging This is when a programmer finds and fixes errors (also known as bugs) in their code.

Event In the computer world, an event is an action that causes something to happen. When you're playing tag and you are "It," if you touch someone, that person becomes "It." Touching the person is an event.

Loop This is when a set of instructions is repeated. When you take turns and follow the rules of a game over and over again, that's a loop.

Perseverance This is when something is challenging, but you don't give up. You keep trying. Learning to ride a bicycle is hard. You lose your balance a lot. You might fall down. When you keep trying to learn, even if it is hard, you have perseverance.

Sequence This is the order in which things happen. The routine you have for getting ready for school in the morning is a sequence. For example, you might write your sequence for getting ready like this: Wake up. Get dressed. Eat breakfast. Brush teeth. Walk to school bus.



Everyday Algorithms

You can write an algorithm for almost anything you do. For example, an algorithm for brushing your teeth might start with “pick up toothbrush, pick up tube of toothpaste, take cap off toothpaste,” and so on.

Try writing algorithms for things you do every day!

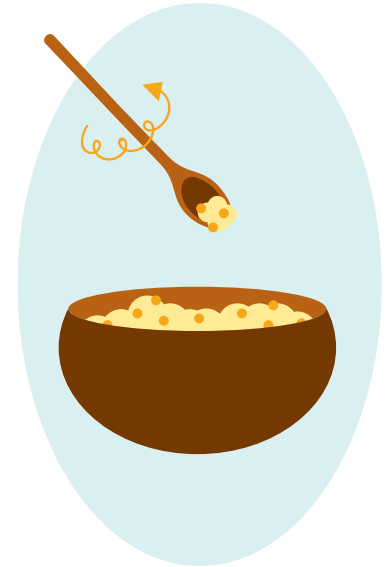
- What’s your algorithm for getting ready for bed?
- What’s your algorithm for making your favorite sandwich?
- What’s your algorithm for changing the water in your goldfish bowl?

STEP 2 Use loops to improve your algorithm

When you make a cake, you mix the ingredients by stirring—a lot!

When you repeat the same action over and over again, that’s called a **loop**.

Programmers use loops to tell computers to repeat steps of their programs. Loops make code shorter, easier to write, and easier for a computer to understand.



STEP 3 Use events to make things happen

When you bake a cake, you set the oven timer.

When it goes off, what do you do? You check to see if your cake’s done!

When that timer goes off, that’s an **event**. The buzz makes you check on your cake.

Programmers use events in their algorithms to make something happen. For example, a video game character might open a door, then jump to a new level. Opening the door is the event that makes the character jump.

STEP

4 Learn about women in computer science

Women have built computers, written programs, and imagined new and different ways for people to use computers.

Women wrote computer programs to send astronauts to space, to do hard math problems in a split second, and to design huge ships in just a few hours.

You can learn to be a computer scientist today! How could you use computers to change the world?

Computer Pioneers: GRACE HOPPER

Grace Hopper was born in 1906 in New York City. Ever since she was a child, she was interested in math and engineering. When she was 7, she took apart alarm clocks to see how they worked!

Grace wanted to study math and science in college but didn't get in the first time she applied. She showed perseverance, applied to college again, got in, and earned a PhD in mathematics from Yale.

Grace joined the US Navy Reserve during World War II. She worked with a team to create the first electronic computer in the US, called the Mark 1. The computers Grace developed were as big as entire rooms!

She also created the first computer language that helped computers understand words, not just numbers.

She also imagined that someday people would use computers all the time and that we'd be able to hold them in our hands. She was right!

Grace always wanted people to think of new ways to do things. She had a clock in her office that ran counterclockwise, or backwards, to remind people that there are lots of ways to do things.



Music Machine

In the 9th century, three Iraqi brothers invented the first programmable machine: a flute that could play music on its own. Air could travel through the flute and make sounds come from a pump, not a person.

Hundreds of years ago, this flute was described in a famous book called *The Book of Ingenious Devices*.



STEP 5 Create your own set of commands that use events

In computer code, events make things happen.

In the real world, events also make things happen.

- ▶ When the oven timer goes off, you check your cake.
- ▶ When the cake begins to cool, you take it out of the pan and put it on a rack.
- ▶ When it's completely cool, you frost it.

Can you create a set of commands that includes events?

Remember, different events will trigger people (or computers) to do different things.

Add events to finish this story:

Today, I met a big dog on the way to school.

He _____ **(verb)**, and I laughed out loud.

I _____ **(verb)**, and the dog sniffed my hand and backpack.

The dog followed me to school. When I got to school,

I _____ **(verb)**, and the dog wagged his tail and barked.

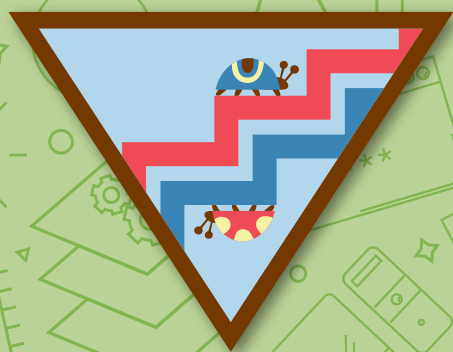
I told my teacher about the dog. She came outside to see and noticed he had a collar with a tag.

She _____ **(verb)** his owner. I waited with the dog until his owner came to get him.

**Now that I've earned this badge,
I can give service by:**

- **Sharing what I've learned about Grace Hopper to inspire other girls to code.**
- **Doing a show-and-tell at school on how to create an algorithm for an everyday task.**
- **Encouraging others to persevere when projects are difficult.**

I'm inspired to:



Badge 2: Digital Game Design

Did you ever wonder how your favorite video game was created? Learn how people create new games—and find out how games can also help people to learn new skills and experience new things.

Steps

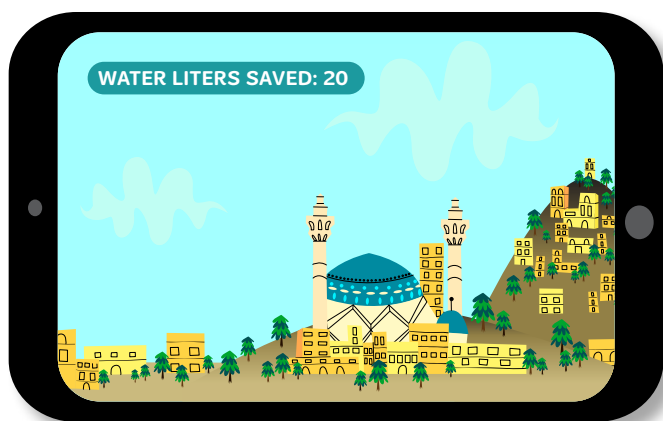
1. Discover how game design can be used “for good”
2. Explore tools used to develop digital games
3. Plan a maze game
4. Build, test, and improve your maze game using iteration
5. Share your game with others

Purpose

When I’ve earned this badge, I’ll know how to think like a game maker. I’ll know how to use iteration to plan, build, and test my game.

STEP 1 Discover how game design can be used “for good”

A game about living in the desert can teach people how to save water. Playing another game can give rice to hungry people. Video games aren’t always just for fun—they can make a real difference. What kind of problem would you like to solve with a video game?



STEP 2 Explore tools used to develop digital games

Computers follow commands that programmers write for them. Those commands are called algorithms.

When you make a digital game, you create algorithms that tell the game’s characters what to do. An algorithm will tell the game characters how to move. Another algorithm might give the character choices.

Sequence, loops, and events are all important parts of the computer programs that make video games.

Words to Know

Digital games

These are games you can play on your phone, computer, TV, tablet, or digital gaming machine. They’re also called video games.

Iteration

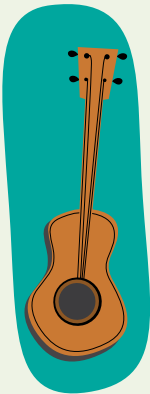
This is when you do something many times to make it better. Think about drawing a picture of a flower. You may draw it once, then decide to add leaves. So, you draw it again with leaves. Each time you draw the flower, you’ll make it a little better.

The School of Games

In the past, if you wanted to learn something new, you'd read a book or ask a teacher.

Now, you can play a game! Video games can teach you how to play the guitar, fly an airplane, plan a city, or build spaceships.

What would you like to learn how to do? Decorate cupcakes? Dance hip-hop or salsa? Train your dog to do tricks? You could invent a game to teach others how to do that, too!



STEP 3 Plan a maze game

Big ideas need big plans. Imagine you were going to design a new Brownie uniform.

- ▶ First, you'd ask other Brownies what they'd like the new uniform to be like. Do they want pockets? Do they like buttons or zippers better?
- ▶ Then, you'd imagine what the uniform would look like.
- ▶ Next, you'd make sketches of your uniform design.
- ▶ You'd show the sketches to some Brownies and ask for feedback.
- ▶ Then, you'd make more sketches to improve your design.
- ▶ When you've finished improving your sketches, you'd make a uniform out of fabric.
- ▶ You'd share your new uniform with the other Brownies and ask for feedback.

You just went through the design process by following these steps: Ask, Imagine, Create, Improve, Share.

You can also use this process to make a good video game.

THE DESIGN PROCESS

Ask



Imagine



Create



Improve



Share

STEP 4 Build, test, and improve your maze game using iteration

As game makers develop new games, they play them throughout the design process to see how they work. This is called **playtesting**.

Playtesting shows game designers where they can make the game better. Every time they test and improve their game, they use **iteration**. Iteration is when you repeat a process many times. When you practice a song on the piano and correct your mistakes each time you play it, that's iteration.

When you create a great video game by testing and improving it, you're using iterative game design.



Play like a Computer

Think about your favorite game. How do you play it?

The rules for how you play the game is an **algorithm**. For example, how do you take a turn? Do you roll the dice or spin a spinner to move your game piece?

The order you do the steps is the **sequence**: First, you roll the dice. Then, you move your piece that number of spaces.

If you take turns with other players, that's a **loop**.

Can you figure out an algorithm, sequence, and loop for your favorite game?



STEP

5 Share your game with others

How can you find out if your game works? Ask a friend to play it!

Computer scientists and game makers show their work to other people to get feedback. You can help other programmers by playing their games and seeing what works and what doesn't. Plus, playtesting other people's games might give you ideas about how to make your game better.

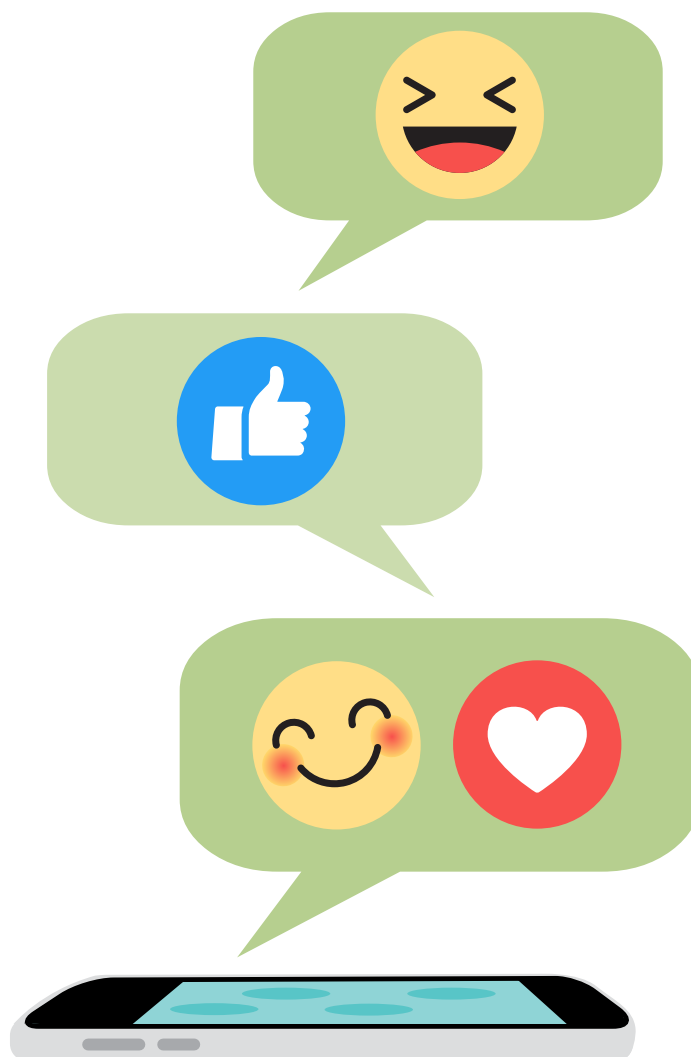
The best part about sharing your game with others is that it's FUN!

Girl Game Designers

You love to play video games, so why not design them, too? Women have designed some awesome games, such as:

- A video game based on a Japanese pop-up book
- A game where you figure out how to turn the lights back on when they all go out
- A game that makes a puzzle out of one of your photos
- A puzzle game where players have to rescue the prince

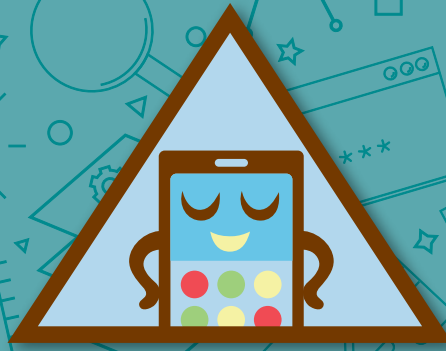
They were all designed by women! You could be next!



**Now that I've earned this badge,
I can give service by:**

- **Telling other people about games they can play to help charities or science.**
- **Teaching others about the design process game designers use to make video games.**
- **Using iteration to make something better.**

I'm inspired to:



Badge 3:

App Development

Computers can help us do things big and small, like send rockets to the moon or get directions to a friend's house. The programs on your phone or tablet are called applications or apps.

Learn how programmers break down the big challenge of developing an app into smaller steps. Then, design your own idea for an app that solves a problem for someone else.

Steps

1. Discover how apps can be used “for good”
2. Decompose the needs of your app user
3. Design your app screens
4. Create algorithms for your app that include events
5. Share and improve your app with user feedback

Purpose

When I've earned this badge, I'll know about user-centered design and the process computer scientists use to develop apps.

STEP

1 Discover how apps can be used “for good”

Teams of programmers are coming up with apps that help people every day.

Animal shelters and animal rescue groups use apps to help dogs and cats get adopted. People who are far apart use apps to stay in touch with video chats or written messages. Apps help people turn their exercise (biking, running, walking the dog) into donations to charities.

If you could create an app to help others, what would it do?



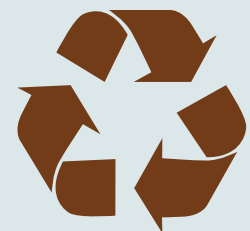
Green Apps for a Green Planet

Do you want to help protect plants and animals? Do you want to have clean air and water? Apps can help you protect our world from pollution.

Earth-friendly apps can help people:

- Learn how to recycle trash
- Buy things made out of recycled materials or buy things that don't have lots of plastic wrappers
- Find a farmers' market to buy fruit and vegetables that were grown nearby
- Ride a bus or train, or join a carpool, to use less gasoline and create less air pollution

These apps help people take care of our earth.



WORDS TO KNOW

App App is short for application. This is a software program that runs on your computer, tablet, or phone. Apps can be entertaining, like when you play a game or watch a movie. They can be helpful, like giving you directions from your house to the soccer field. They can also teach you something new, like a language you want to learn.

App features These are the parts of an app. They could be things like using the camera, a welcome video, a help page, or a way for app users to connect with friends.

Decomposition This is when you break down a problem into smaller steps or pieces to solve.



Development This is when you create something new. When you develop something, you create a plan before you begin building. Then, after you build it, you test to see how it works and find ways to make it even better!

User needs These are what potential or current users need to solve the problem your app is meant to solve.

User-centered design When programmers create an app, they try to include their user at many stages of the development process. For example, they'll talk to their user before they start planning or have the user test the app once it's built. This helps programmers develop apps that are easy to use and helpful to others.

STEP 2 Decompose the needs of your app user

When you make a plan to do something, you think about the steps to do it. You can solve problems the same way! If you take a big project, like building a snow fort, and break it into smaller steps, it makes it easier to do.

- ▶ You call some friends to help you.
- ▶ You gather shovels and buckets to use to make your fort.
- ▶ You put on warm clothes to go out in the snow.
- ▶ You meet your friends outside and choose a place for your fort.
- ▶ Some of you dig in the snow with shovels to make a floor for the fort.
- ▶ Some of you make snow bricks by filling the bucket with snow and then tipping it upside down.
- ▶ You and your friends stack up your snow bricks to make the walls of your fort.

When you break a big problem or project into smaller steps, it's called **decomposition**. When programmers find a problem with a program they've written, they use decomposition to make the problem easier to solve.

How can you use decomposition to design an app?

STEP 3 Design your app screens

Apps for a Better YOU

Some apps have been created just to help kids.

Apps can help kids stay active by challenging them to get moving. The apps might show their character run, jump, or spin every time the kid does.

Digital diary apps give kids questions to write about. This can help them think about how they feel.

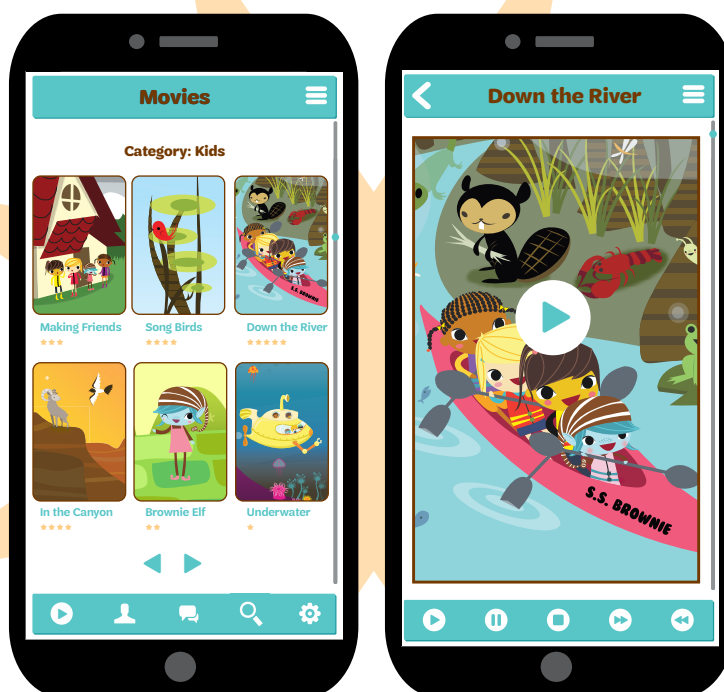
Some apps even help kids understand bullying and friendships.

What other kinds of things do kids need help with?

When app developers have an idea for a new app, they draw it on paper first. They show their drawings to other people and ask for ideas on how the app idea could be better. This helps the programmers to try different ideas before they start coding.

Think of app screens like a storyboard that shows how an app user would move through the app.

For example, if you're choosing a movie to watch on an app, you might first see an app screen showing all the different options for movies and TV shows. Then, you click on a movie, and it appears on the screen for you to watch! If you were to sketch it out, this would be two different pictures: one of the first screen with all the movies and another of the movie playing.



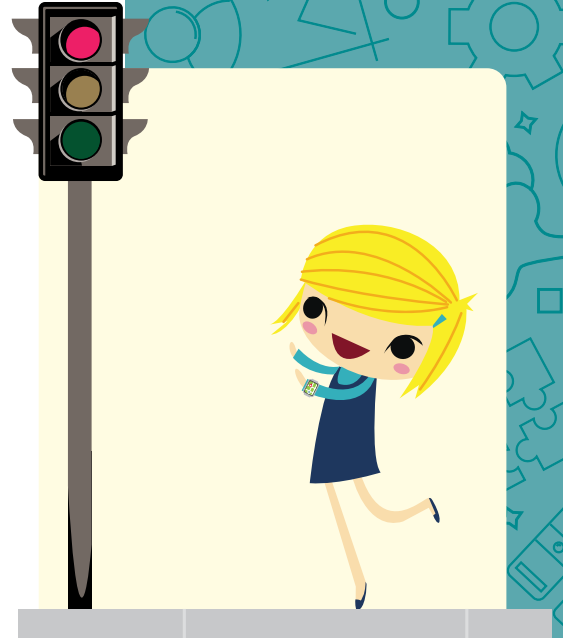
STEP 4 Create algorithms for your app that include events

Algorithms are made up of steps.

An algorithm for going to a friend's house might include steps like choose a time with your friend to meet, get directions to her house, and walk to her house.

You can add events to make algorithms more interesting. Events in the algorithm might be “stoplight turns red” (so you have to wait to cross the street) and “knock on her door” (and wait for her to answer).

When app developers test their apps, they might also have to find and fix errors in their programs—this is called **debugging**.

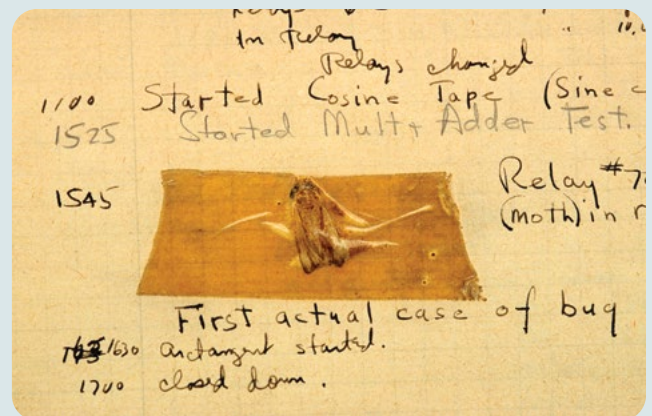


There's a Bug in My Computer

In the early 1900s, Thomas Edison used the word “bug” to describe problems with his inventions. Grace Hopper was the first person to use the word “bug” about computer programming.

When she was working on the Mark 1 computer, the computer kept making mistakes. Her team opened up the computer to see what was wrong and found a moth. They taped it into a log book where they wrote down all the things that happened with the Mark 1.

When Grace saw the moth—a real bug, not just a mistake in Mark 1's program—she wrote, “First actual case of bug being found” next to it in the log book. That's why people today still say they need to “debug” their computers!

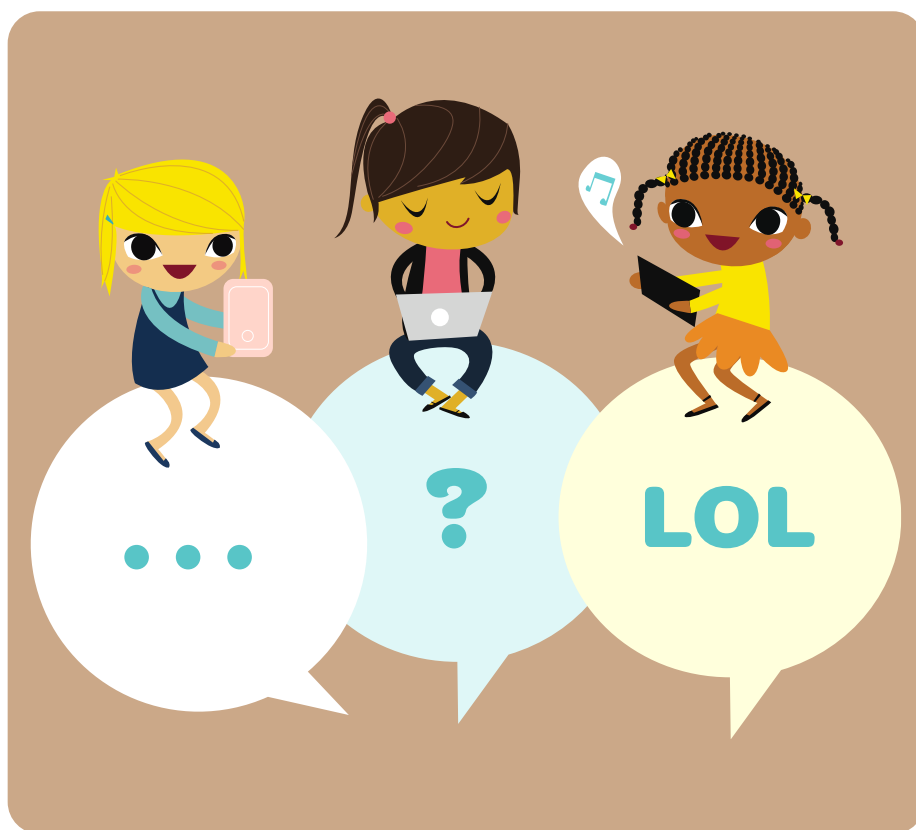


STEP

5 Share and improve your app for user feedback

What's better than coming up with an idea, working hard to make it great, and then sharing it with people? When you show your app to others, they try it out and give you feedback.

When someone else uses your app, you'll find out if it works the way you want it to. The tester's feedback can help you make your app even better. You can use their ideas to make improvements to your app.



**Now that I've earned this badge,
I can give service by:**

- Explaining where “computer bug” comes from.
- Teaching others how to solve big problems by breaking them into smaller parts.
- Telling my friends or family what I learned about coding.

I'm inspired to:



Made possible by a generous grant from Dell

© 2019 Girl Scouts of the United States of America.

All rights reserved. No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or by any other electronic or mechanical methods now known or hereinafter invented, without the prior written permission of Girl Scouts of the United States of America, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permissions requests, write to Girl Scouts of the United States of America at the address below or visit the www.girlscouts.org website to access permission request forms.

Links to third-party websites are provided for convenience only. Girl Scouts of the USA (GSUSA) does not endorse nor support the content of third-party links and is not responsible for the content or accuracy, availability, or privacy/security practices of other websites, and/or services or goods that may be linked to or advertised on such third-party websites. By clicking on a third-party link, you will leave the current GSUSA site whereby policies of such third-party link may differ from those of GSUSA.

First published in 2019 by Girl Scouts of the USA
420 Fifth Avenue, New York, NY 10018-2798
www.girlscouts.org

Printed in the United States

Grace Hopper: UAV 605.2701 (U822); olvwork286535; Harvard University Archives; Computer bug: Division of Medicine and Science, National Museum of American History, Smithsonian Institution

UPC 64047



7 31955 64047 7