

Beavers Digital Maker Stage 1

Andrew Oakley andrew@aoakley.com April 2017 Public Domain www.cotswoldjam.org

Based on numbered requirements from:

<https://members.scouts.org.uk/supportresources/4283/digital-maker-staged-activity-badge>

Proposed date Fri 12 May 2017, 3rd Tewkesbury Beavers

Plan is to have three groups each doing one activity per 20 mins, then rotate groups.

Requirement 1: Show that you can identify a computer, the basic components inside a computer and what their purpose is.

Staff member: Andrew Oakley

Activity & Materials: Connect up and turn on a Raspberry Pi computer, including:

- Raspberry Pi
- Raspberry Pi power supply (this is low-voltage and safe)
- Monitor
- Monitor power supply (CAUTION, this is mains voltage; consider adult assistance)
- HDMI cable
- SD card
- Mouse
- Keyboard

For the ultimate in safety, suggest you connect everything to an unplugged mains extension lead, and have an adult turn on the extension lead only when the computer is fully connected.

Explain volatile and non-volatile memory. Compare memory to storage.

- Memory (volatile) forgets everything when it is turned off. Show the RAM (Random Access Memory) chip on the bottom of the Raspberry Pi.
- Storage (non-volatile) remembers what it's storing even after being turned off. Show the SD card on the Raspberry Pi. Other computers use "hard drives" or "solid state drives".

Take a photo of the group with their working computer

Requirement 2: Show that you can create instructions for something you do every day such as getting dressed in the morning or making a sandwich.

Staff member: Liza Gardiner-Orton

Materials: Lined paper, pencils, erasers

Activity: Write down a list of steps to make a breakfast comprising:

- Orange juice
- Cereal
- Toast

Imagine you have an android (a robot shaped like a man, like C3PO) in your kitchen and you want him to make breakfast for you. Write down your list of instructions! This is a program.

Think about:

- Where will you get the ingredients from? Cupboard, fridge, shelf? Do you need to open any doors? Don't forget to close the doors!
- Will you need cutlery, bowls, cups or plates? If so, where will you get them from? Will you get everything all ready first, or just get items when you need them?
- Are some things inside other things? For example, milk will be inside a bottle which is inside the fridge. Don't forget to open the containers!
- What will you have with your toast? Margarine, marmite, jam? How will you spread it?

Trivia:

Notice how you do some jobs several times. For example you might "walk to the fridge and open the fridge door" five or six times. These groups of instructions are called "subroutines".

Notice how getting lots of things ready at the same time, can make the list of instructions shorter. For example, if you get everything you need from the fridge in one go, and put them all on the table, it saves you walking to and from the fridge each time, and opening and closing the door each time.. This is an example of making programs more efficient (faster).

Do you need to instruct the robot to move his hands and feet individually, or do you think he will understand more complex instructions such as "walk" and "take"? There's a name for this. "Low level" programming means you have to go into lots of tiny details. "High level" programming means that the computer or robot already understands the basics.

Notice what you forgot. Did you remember to put the lid back on the jam? Perhaps there is a wasp in your jam now! Changing your program to correct mistakes is called "debugging".

The woman who invented computer programming languages, US Admiral Grace Hopper, invented the word "debugging" when she found a real insect stuck inside her computer, that was causing her programs to go wrong. She started blaming all her own mistakes on "bugs"!

"Android" is the word for a robot shaped like a man. A robot shaped like a woman is called a "gynoid".

Requirements 3 & 4: Design and prototype a game

Staff member: Bill Gardiner

Materials: 2cm squared paper (see next page in document), pencils, eraser, felt-tip pen and a small toy figurine (such as a Lego minifigure or wargame soldier).

Activity: Create a maze game on paper, and then play it.

- Draw the walls of your maze in pencil. Use the sides of the squares to help you.
- Where is the start?
- What is the goal of your maze game? Do you have to reach the middle, or the end? Do you have to collect one or two items, or pass over special squares? Does it matter how many steps you take?
- What moves can the player make? Forward, backwards? One square at a time, or lots? Do they turn (spin) left or right (so they face a new direction), or do they sidestep (so they always face the same way)?
- What are the other rules? What happens if you touch or crash into a wall - does the player go back to the start, does he lose a life (how many lives does he have?), or does it not matter?
- Make sure it is possible to win your game! For example make sure the player can get from the start to the goals - use the eraser to rub out any bad walls.
- When your maze is finished, use the felt-tip marker to go over the pencil lines to make it easier to see.

Now explain to your friend how to play your game, and then play it.

- What worked well?
- What would you do different if you did it again?

My Maze Game - Name: _____

