The Design Process

A supplemental lesson for Firia Labs CodeX - AP CSP

Review the remix project

When you created your remix project, you were asked to follow 5 steps:

Let's review them

Review the steps for the remix project

Step #1

Review previous programs to remember what you already know how to do.

Step #2

Come up with a new programming idea

Step #3

Plan the solution to your idea: what it will look like (buttons pressed) and coding

Step #4

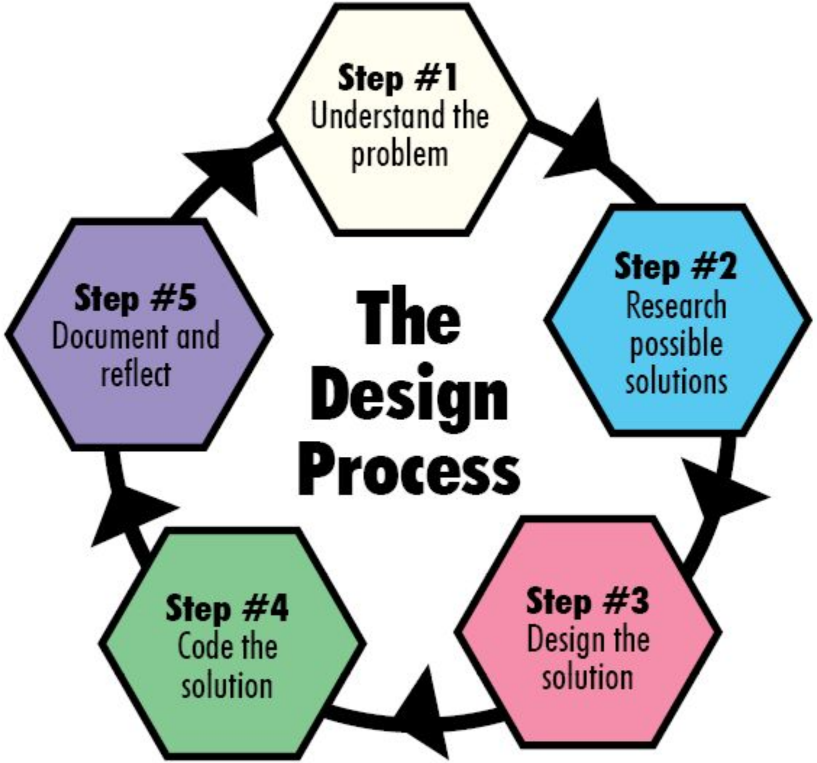
Code the solution.

- A few lines at a time
- Documenting and fixing bugs

Step #5

Get feedback on the project and improve it

The Design Process



Step #1
Understand the problem

Step #2
Research possible solutions

Step #3
Design the solution

Step #4
Code the solution

Step #5
Document and reflect

The Design Process

You were basically following the design process!

Many industries, including computer science, use a design process when working on a new project.

Definition: The design process is a tool that helps you break down large projects into smaller, easier-to handle stages

We just re-arrange the remix step 1 with step 2:

Step #1

Come up with a new programming idea that solves a problem – make sure you understand what it should do

Step #2

Review what you already know how to do. What programming skills and concepts can you use?

Step #3

Plan the solution to your idea: what it will look like (buttons pressed) and coding (variables, conditions, etc.)

Step #4

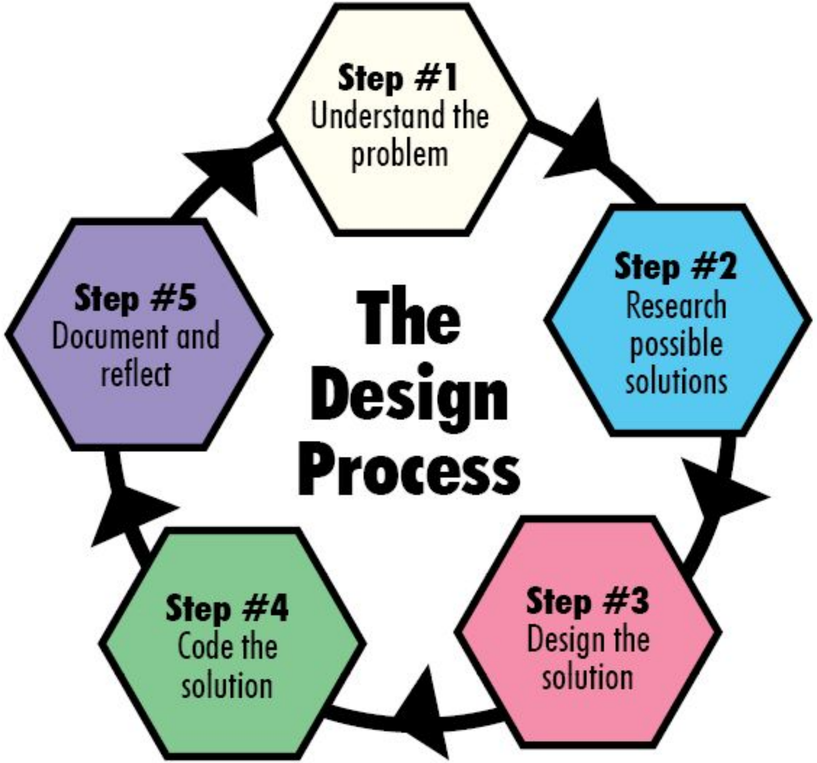
Code the solution.

- A few lines at a time
- Documenting and fixing bugs as you go

Step #5

Document your code. Get feedback on the project, and reflect on how it works so far. Then improve it

The Design Process

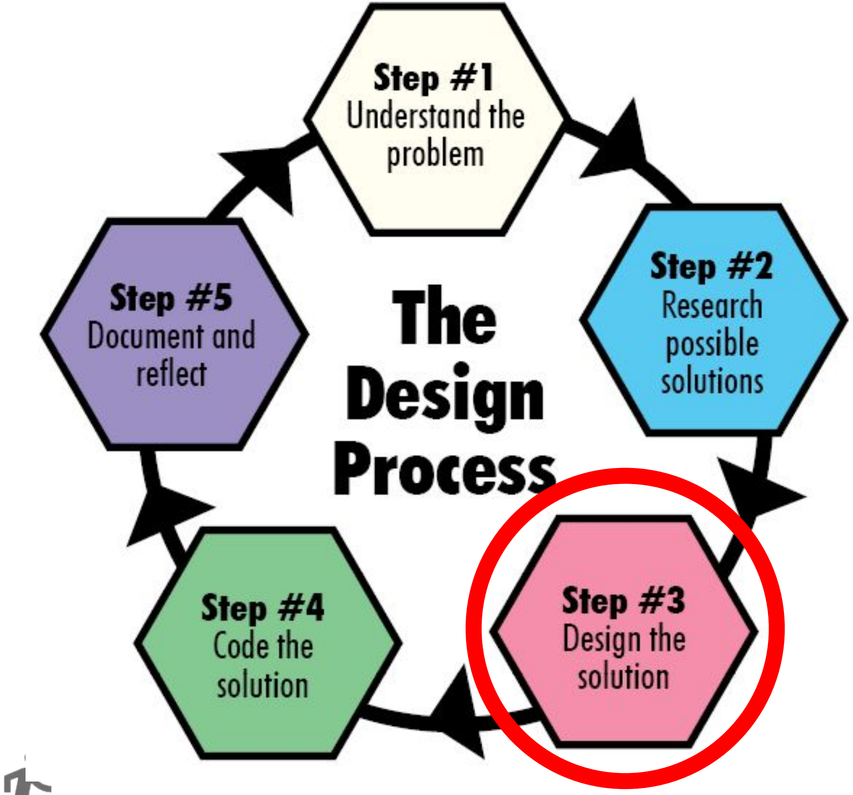


As you can see, the design process is really a cycle

- Programmers frequently improve and modify their code
- After each complete cycle, a new problem may arise or idea that can add to the functionality of your program
- Then you start a new cycle all over again
- Each time you go through the cycle, you have completed one iteration
- Iteration is like repeating: you repeat the cycle as many times as needed

FIRIA LABS

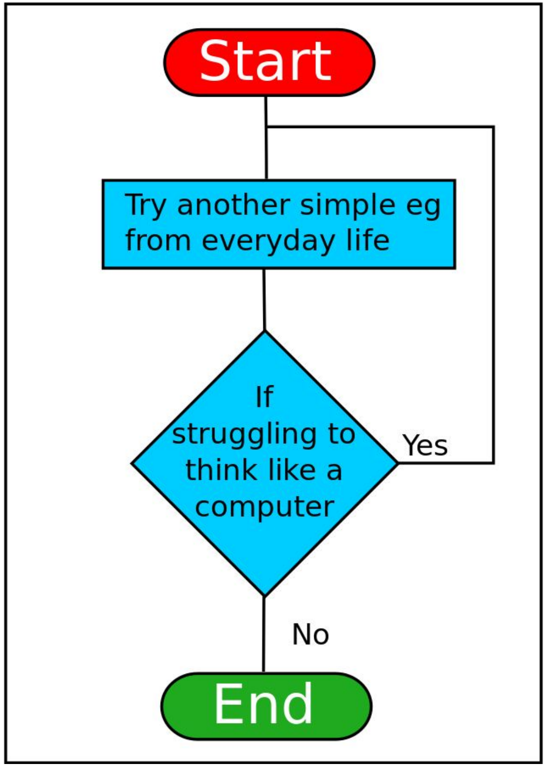
Step #3 - Design a solution



In this lesson, we are going to focus on Step #3 - designing a solution

- For the remix, you used several tables to help with designing
  - Variables needed
  - Buttons pressed
  - Text displayed
  - Images, pixels and audio files

Flowcharts


Start
Try another simple eg from everyday life
If struggling to think like a computer
Yes
No
End

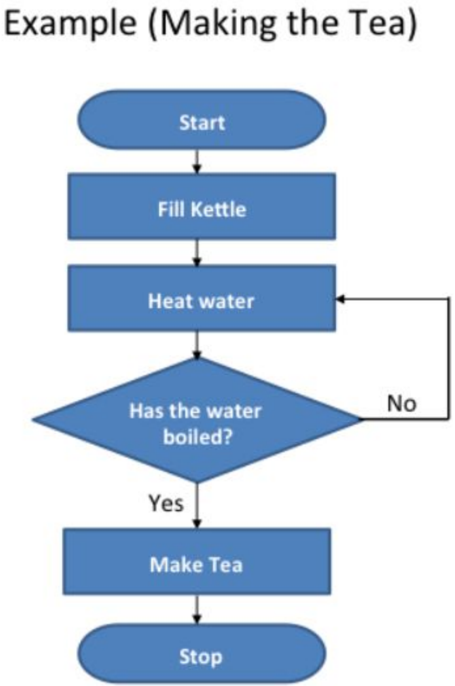
Source: https://en.wikiversity.org/wiki/File:ThinkLikeComputerFlowchart.svg

Another useful tool programmers (and many other professionals!) use is a flowchart. Flowcharts can help you:

- Organize your thoughts and ideas
- Use shapes to visualize the solution
- Track variables
- Identify computations
- Look for patterns

Flowcharts

Flowcharts

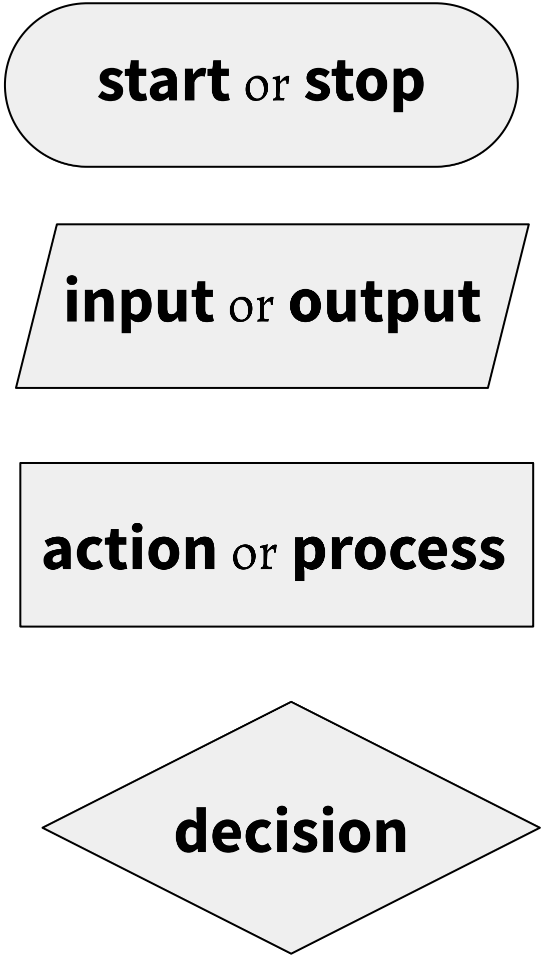


The flowchart visualizes the algorithm, or how to solve the problem.

There are four common shapes used when creating a flowchart:

Source: https://www.teachwithict.com/flowcharts.html#google_vignette

Flowchart symbols



start or **stop**

Use an oval to mark the beginning and end of the program.

input or **output**

Use a parallelogram to show input or output. **Input** could be the button pressed. **Output** could be text on the screen, an image displayed, a pixel lit or a sound played.

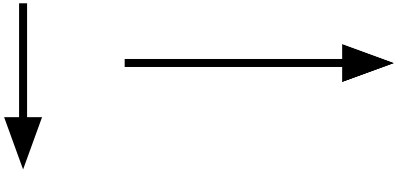
action or **process**

Use a rectangle to process an action. It could be used to assign a value to a variable, or increment a counter, or get a random number.

decision

Use a diamond to make decisions. This shape will have two or more lines that come from it – one for each outcome. This step might ask a question or provide options. The result could be true or false, yes or no, or choices (red, blue, or green).

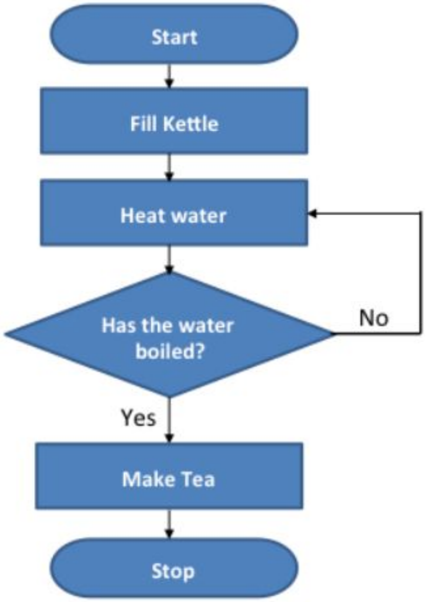
flow lines



Use lines to connect the shapes. The arrows show the direction of the steps. Some lines should include labels, such as yes or no, to explain what is happening.

Flowcharts

Example (Making the Tea)



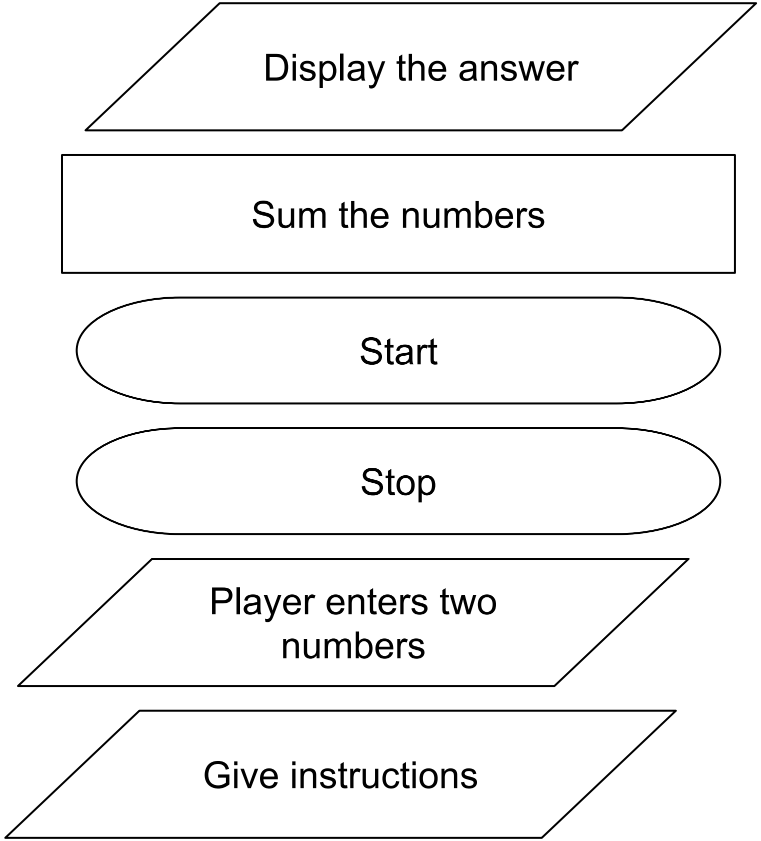
This example doesn’t include any input or output.

- What can you think of to add to this flowchart?
- What are some other examples of how the shapes might be used?

Source: https://www.teachwithict.com/flowcharts.html#google_vignette

Flowchart example

These shapes are mixed up. Can you fix the algorithm by putting the events in the correct order?



Display the answer

Sum the numbers

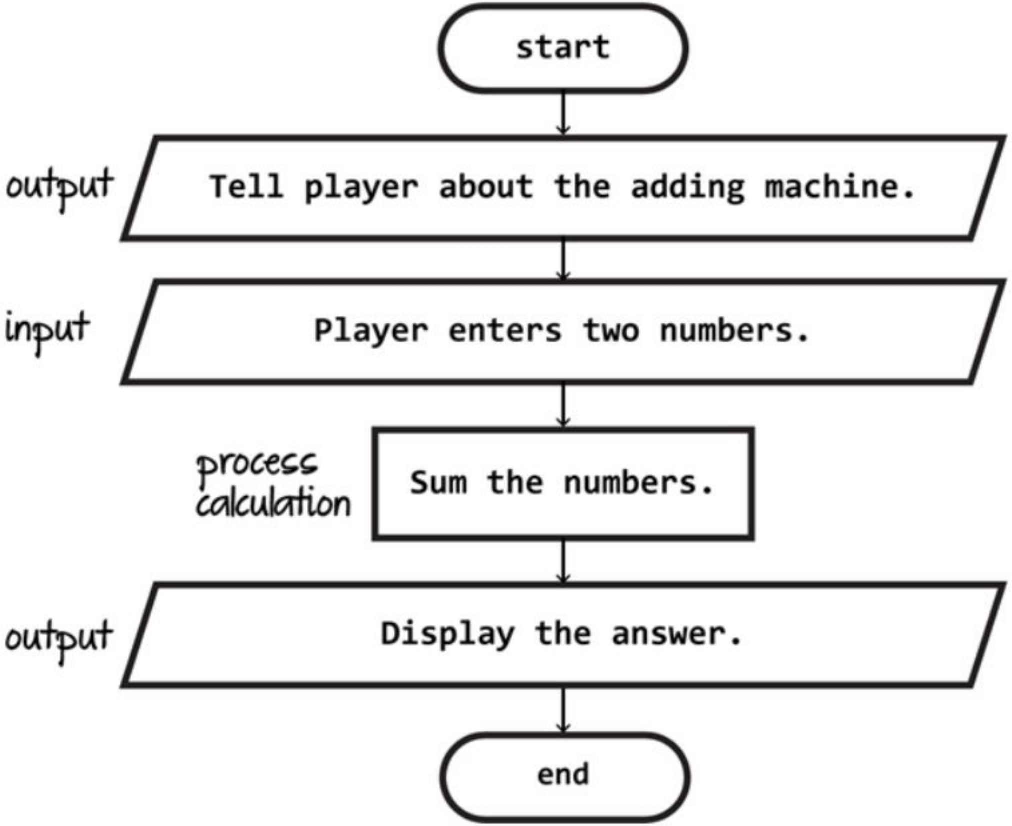
Start

Stop

Player enters two numbers

Give instructions

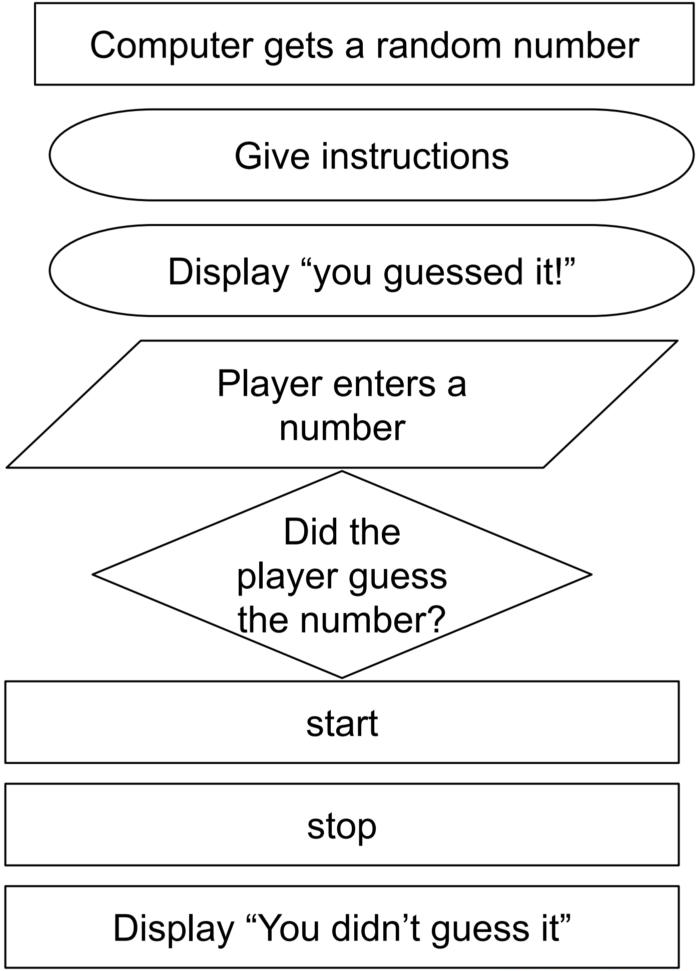
Did you get something like this?



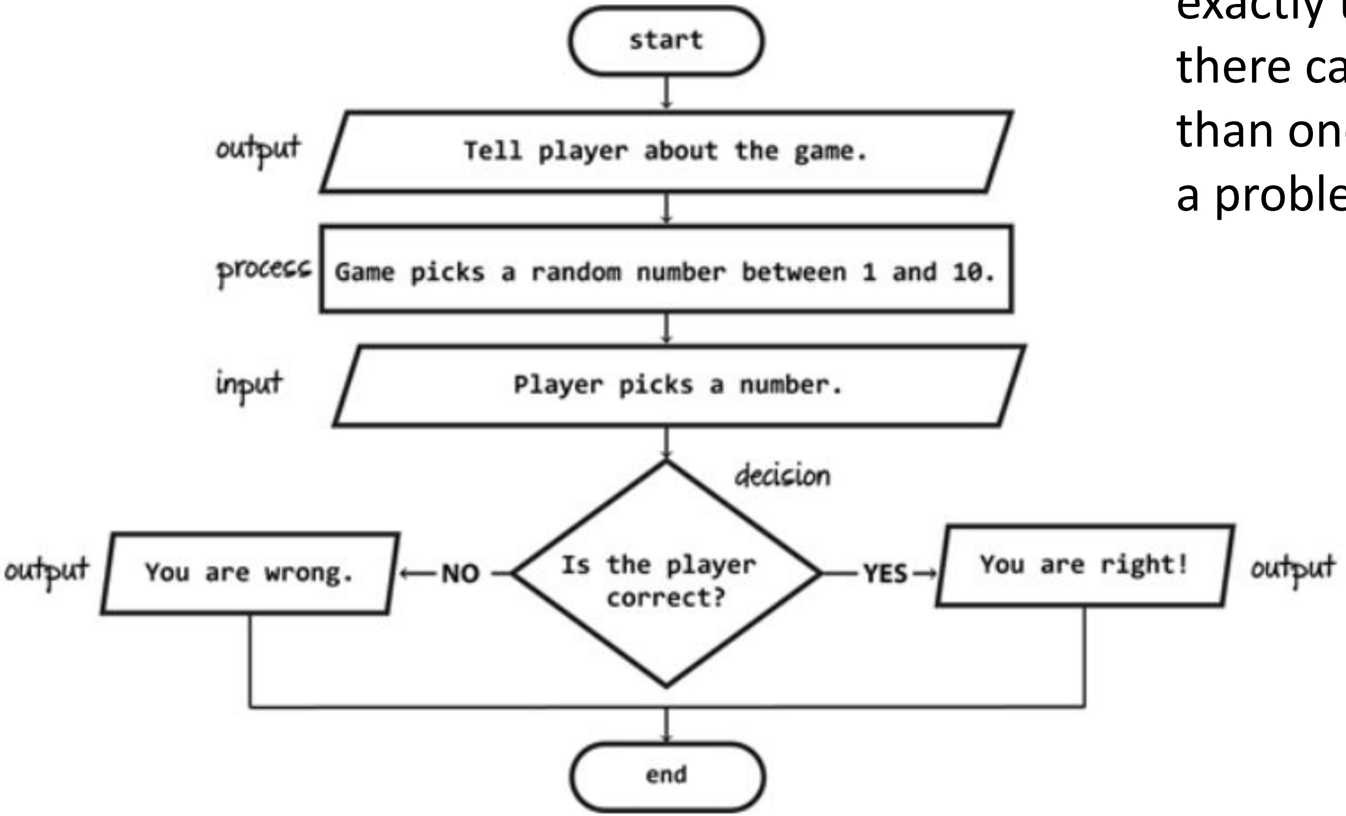
Source: https://blog.technokids.com/programming/how-to-make-a-flowchart-for-programming-easy-to-understand/

Flowchart example

Some of the steps are in the wrong shape, and they are out of order. Can you fix this flowchart?


Computer gets a random number
Give instructions
Display “you guessed it!”
Player enters a number
Did the player guess the number?
start
stop
Display “You didn’t guess it”

Did you get something like this?



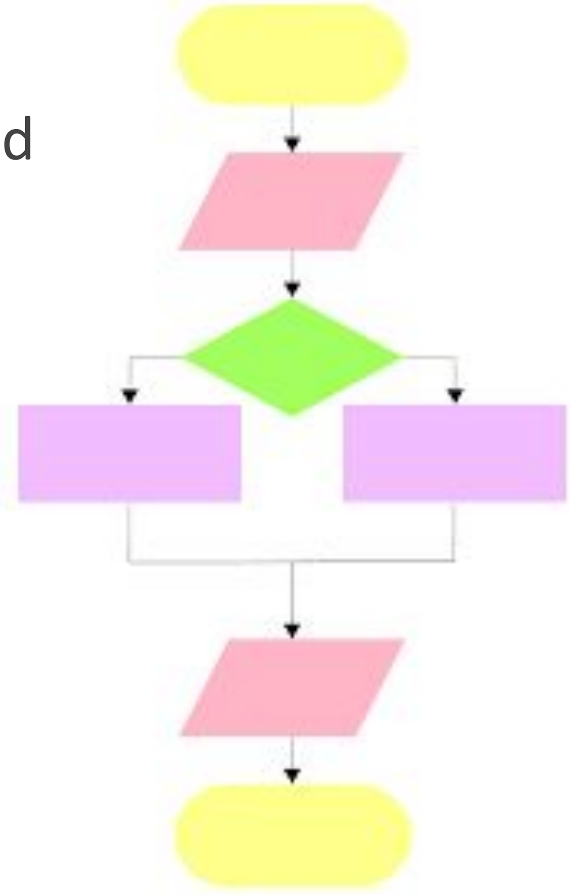
It doesn't have to be exactly the same – there can be more than one solution to a problem!

Source: https://blog.technokids.com/programming/how-to-make-a-flowchart-for-programming-easy-to-understand/

Try one:

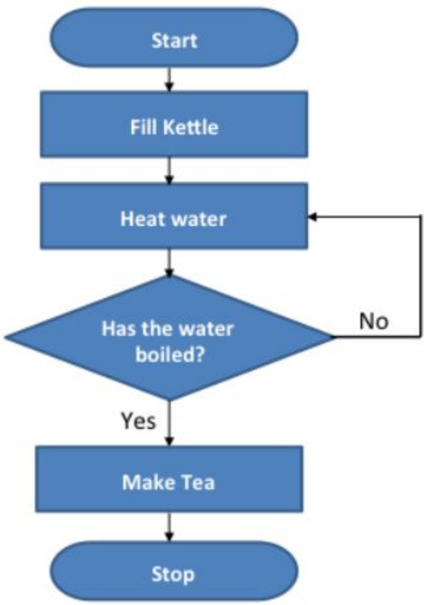
With your partner, select a scenario and create a flowchart:

- Get ready for school
- Select a song to listen to
- Decide what to wear today
- What should I eat for breakfast?
- Do I eat lunch today?



Flowcharts

Example (Making the Tea)

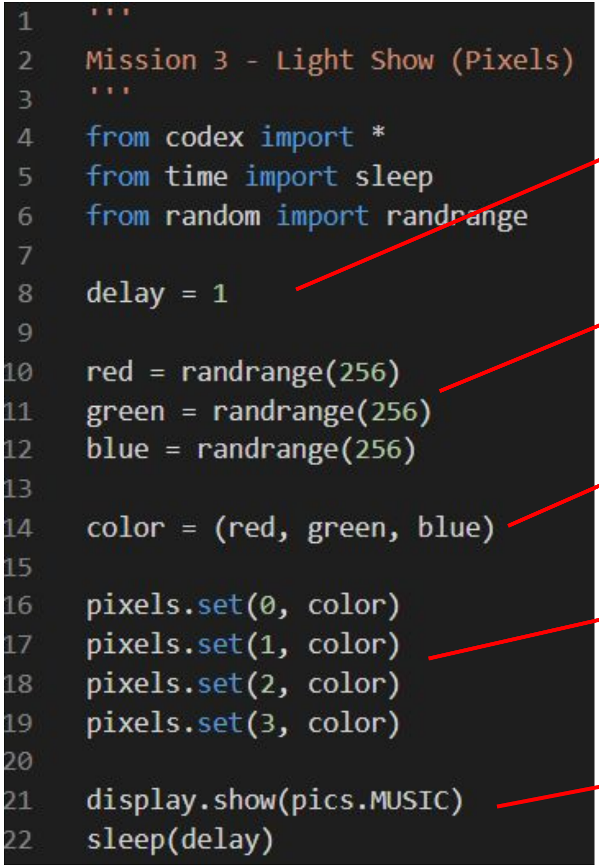


You are now familiar with flowcharts, their shapes, and what each shape is used for.

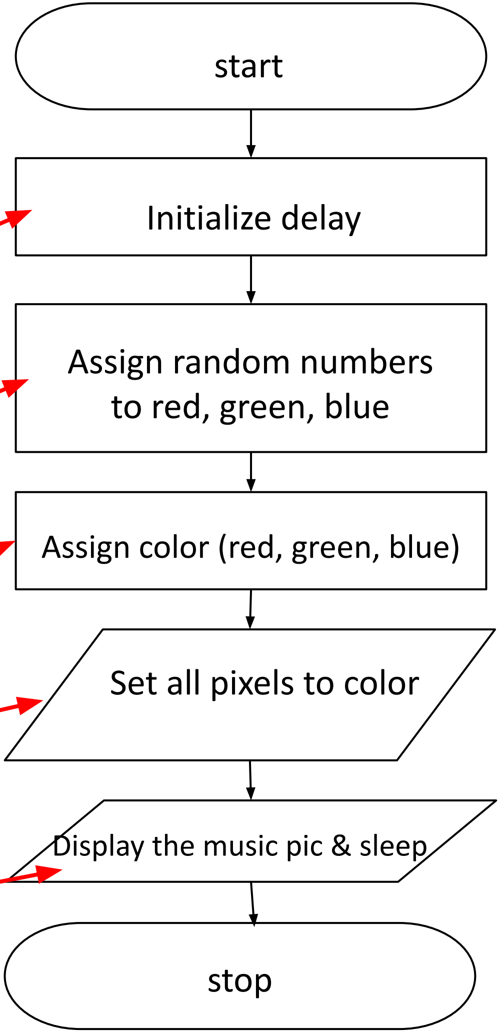
Let’s look at examples in Python programming.

Source: https://www.teachwithict.com/flowcharts.html#google_vignette

Example #1



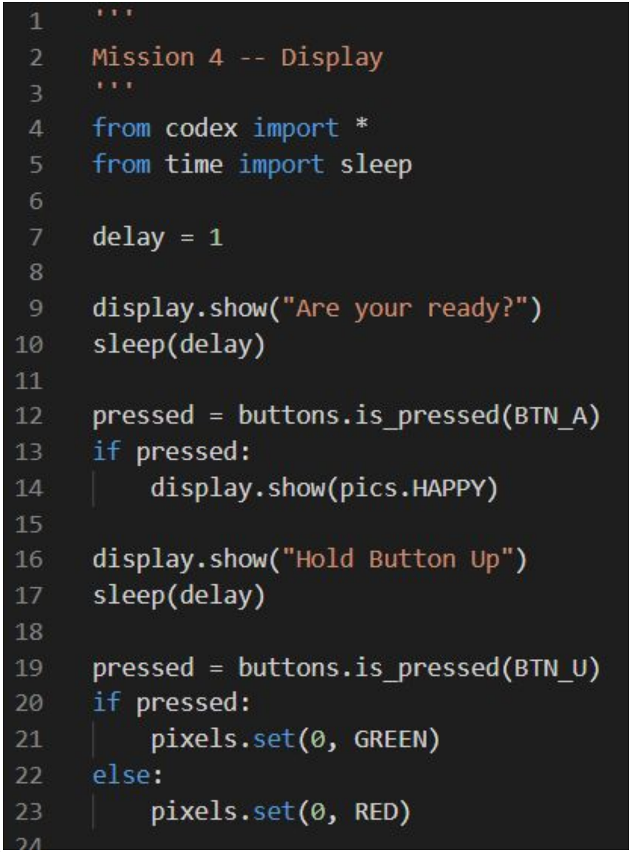
```
'''
Mission 3 - Light Show (Pixels)
'''
from codex import *
from time import sleep
from random import randrange

delay = 1

red = randrange(256)
green = randrange(256)
blue = randrange(256)

color = (red, green, blue)

pixels.set(0, color)
pixels.set(1, color)
pixels.set(2, color)
pixels.set(3, color)

display.show(pics.MUSIC)
sleep(delay)
```



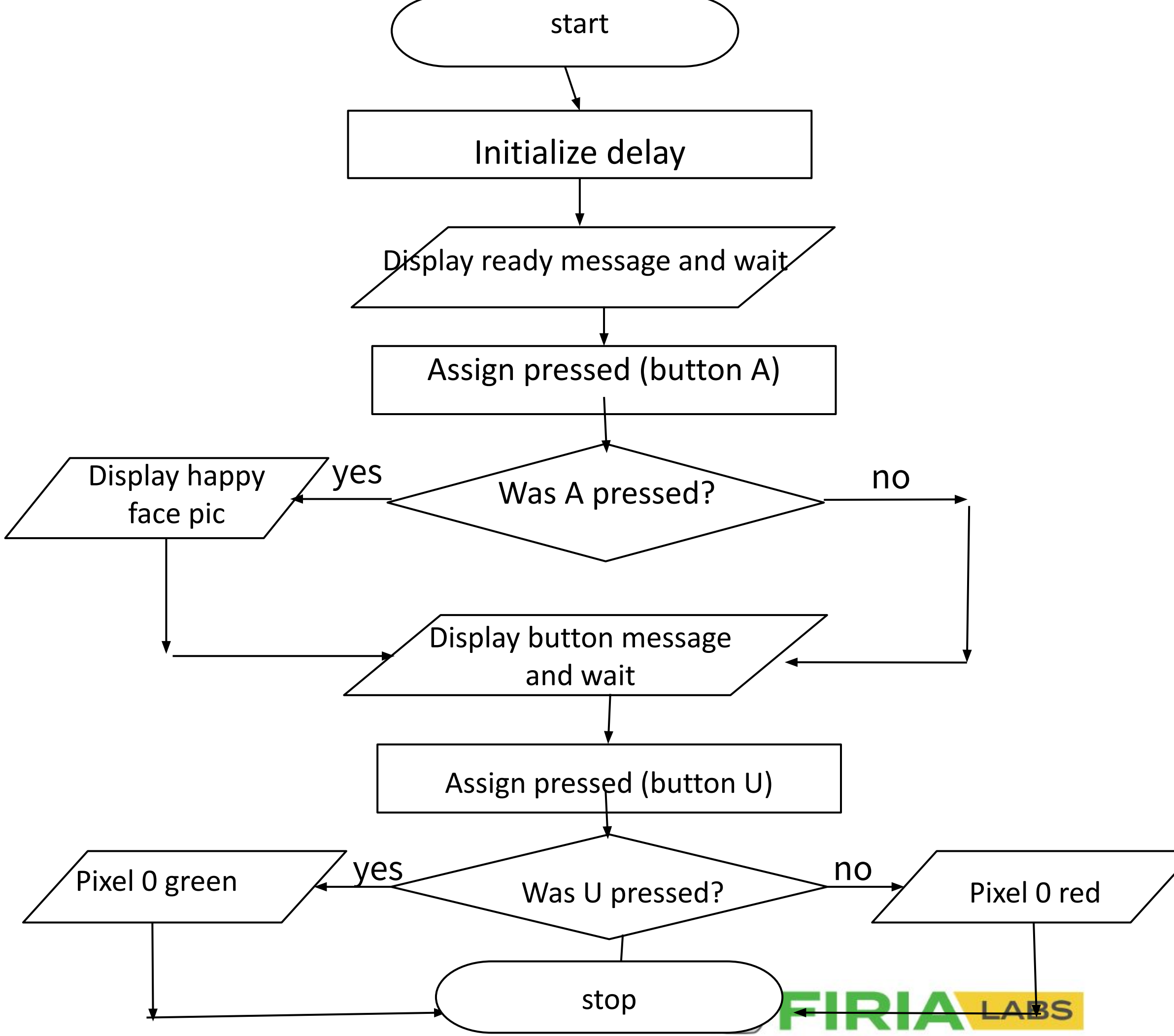
start
Initialize delay
Assign random numbers to red, green, blue
Assign color (red, green, blue)
Set all pixels to color
Display the music pic & sleep
stop

FIRIA LABS

Example #2



```
'''
Mission 4 -- Display
'''
from codex import *
from time import sleep

delay = 1

display.show("Are your ready?")
sleep(delay)

pressed = buttons.is_pressed(BTN_A)
if pressed:
    display.show(pics.HAPPY)

display.show("Hold Button Up")
sleep(delay)

pressed = buttons.is_pressed(BTN_U)
if pressed:
    pixels.set(0, GREEN)
else:
    pixels.set(0, RED)
```



start

Initialize delay

Display ready message and wait

Assign pressed (button A)

Was A pressed?

yes

Display happy face pic

no

Display button message and wait

Assign pressed (button U)

Was U pressed?

yes

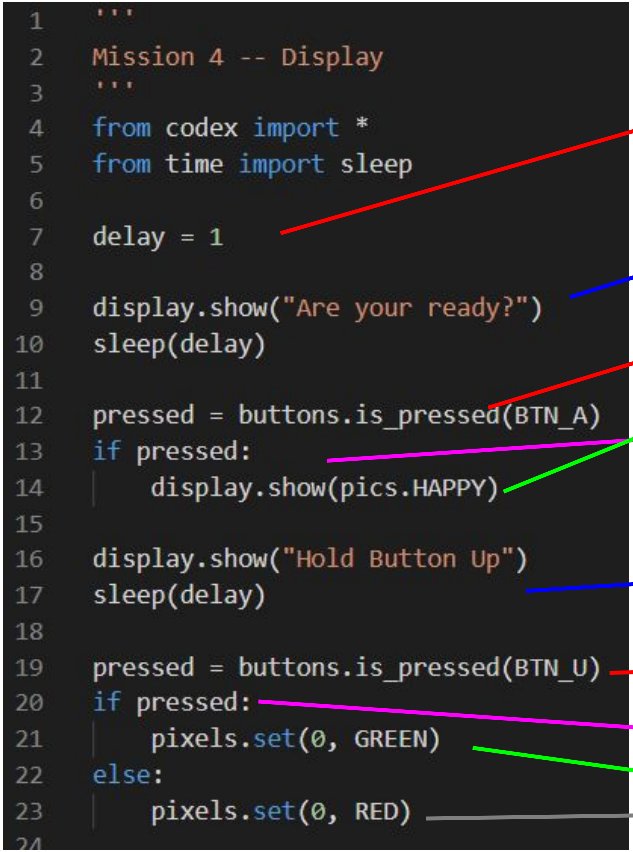
Pixel 0 green

no

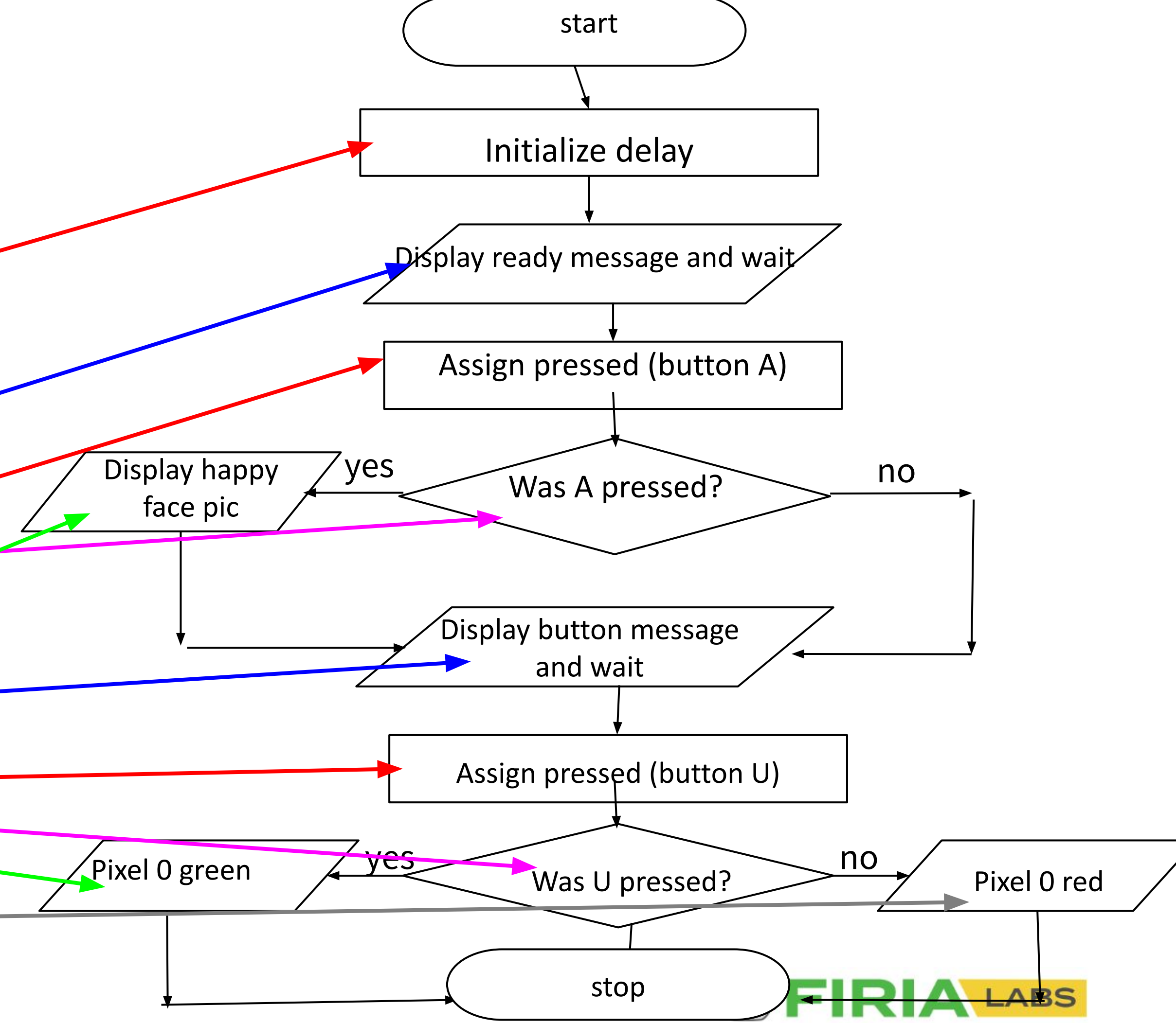
Pixel 0 red

stop

Example #2

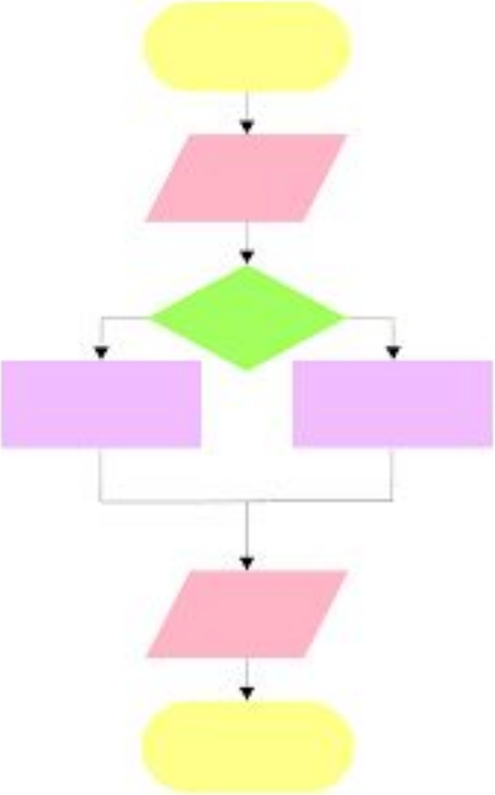


```
'''
Mission 4 -- Display
'''
from codex import *
from time import sleep

delay = 1

display.show("Are your ready?")
sleep(delay)

pressed = buttons.is_pressed(BTN_A)
if pressed:
    display.show(pics.HAPPY)

display.show("Hold Button Up")
sleep(delay)

pressed = buttons.is_pressed(BTN_U)
if pressed:
    pixels.set(0, GREEN)
else:
    pixels.set(0, RED)
```


start
Initialize delay
Display ready message and wait
Assign pressed (button A)
Was A pressed?
yes
no
Display happy face pic
Display button message and wait
Assign pressed (button U)
Was U pressed?
yes
no
Pixel 0 green
Pixel 0 red
stop

Create a flowchart for python code:

Get in random groups of three at the vertical white boards and get sample code from your teacher. Create a flowchart for each one.



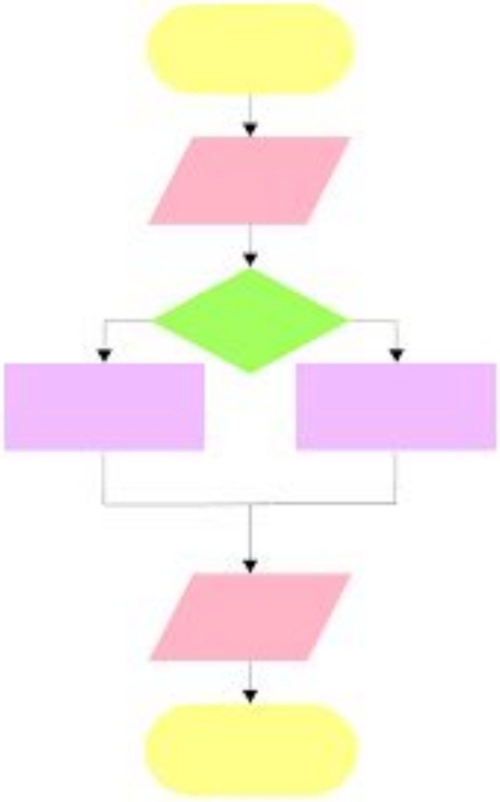
Create a flowchart for python code:

With a partner, open the following programs and create a flowchart for each one:

- Mission 5 – music1
- Mission 3 – pixels1
- Mission 4 – display

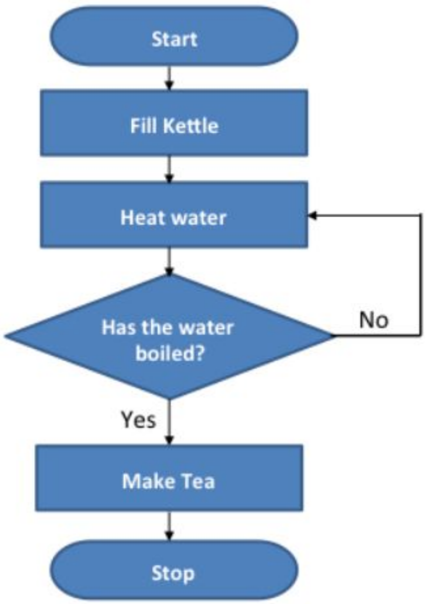
Extension:

- Remix #1



Flowcharts Summary

Example (Making the Tea)



Definition: An **algorithm** is a sequence of steps for completing a task.

Definition: A **flowchart** is a diagram that uses shapes, lines, and arrows to sequence steps.

Definition: A **flowchart** is a visual representation of the input, output, decisions, and actions that take place within a program.

Source: https://www.teachwithict.com/flowcharts.html#google_vignette