

The Engineer's Notebook

An engineer's notebook is like a technical diary!

Engineers use an Engineering Notebook to record ideas, observations, and all work details.

Written ideas

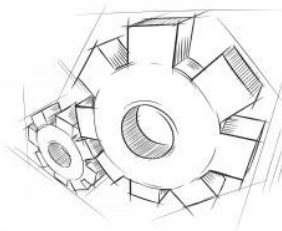
Sketches

Research findings

Technical data

Interview information

Lessons Learned



Scratch

Script

Sprite

Stage

Workspace

GLOSSARY OF TERMS

19

Algorithm

Code Block

Conditional Logic

Features

Requirements

FAVORITE SCRATCH TIPS:



What is an Algorithm?

Examples of using algorithms for everyday tasks:

Write an algorithm to draw a smiley face:



Create your own algorithm!

Task: _____

Steps:

1)

2)

3)

4)

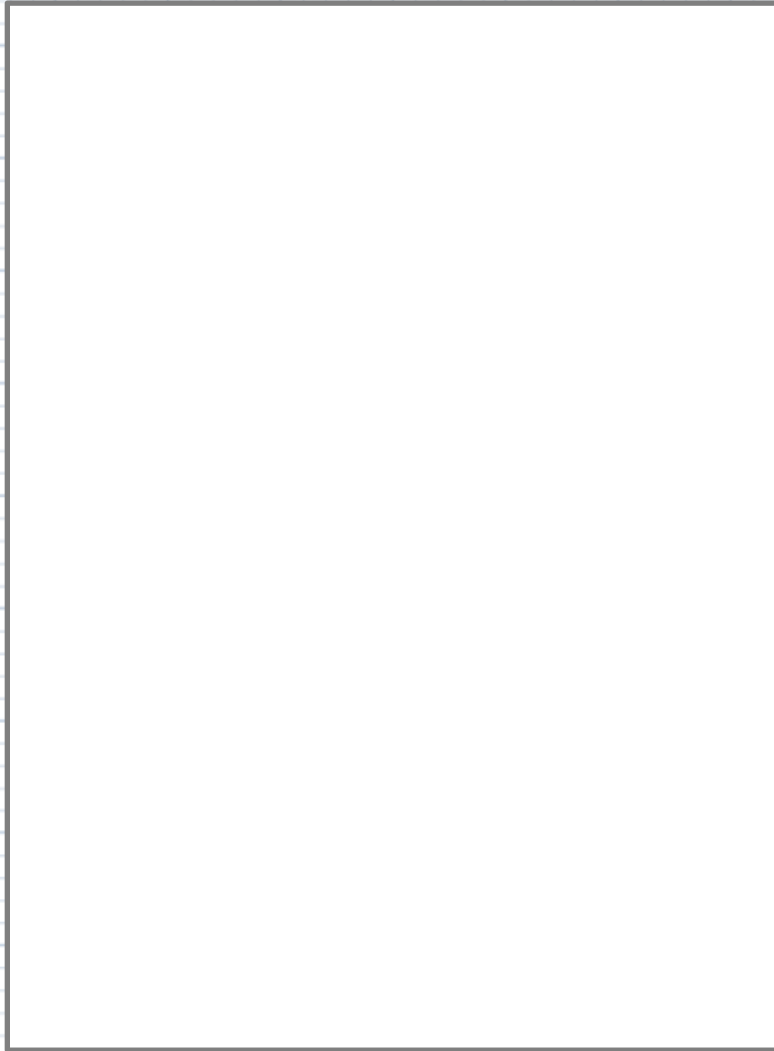
5)

Sell your Game!

Why should someone want to buy your game? What features make it unique? Who is your target audience?

Package your game!

If you were to create the artwork for your finalized game, what would it look like?



Lessons Learned:

What was challenging? Where did you get stuck?

What did you learn? What was your "A-ha!" Moment?

What is Scratch?

Draw your favorite Sprite:



What button makes your program GO: ☐

What button makes your program STOP: ☐

What are the 8 different script types:

(write the name of each script type and color in the box with the matching color)

☐ _____

☐ _____

☐ _____

☐ _____

☐ _____

☐ _____

☐ _____

☐ _____

Complete the cost analysis for your game:

How much does your game cost to produce?

What can you sell your game for, based off of the features you successfully implemented?

What is the profit that will be made off of each game sold?

Write out the instructions for your game
(object of game, how many players, how to win, keyboard and mouse controls, number of levels, etc.)

Lessons Learned:

What was challenging? Where did you get stuck?

What did you learn? What was your "A-ha!" Moment?

An algorithm is a script made up of a sequence of blocks!

What are the 10 blocks?

Describe your final script. What does your sprite do?

Troubleshooting:

What features were too difficult to implement? What made them so challenging? Where did you get stuck?

Outline your algorithm here:

Lessons Learned:

What was challenging? Where did you get stuck?

What did you learn? What was your "A-ha!" Moment?

What are the requirements (goals) of this exercise?

- 1)
- 2)
- 3)

What special features did you choose?

- 1)
- 2)
- 3)
- 4)

Extra Credit: Add conditional logic



Add an "If/Then" or "If/Then/Else" Control block to your code to make certain events happen only if other conditions are met (for example: IF you ball hits a wall, THEN change size/blink/change color/etc.)

Describe your extra credit feature:

List of features would you like to implement in your game:

Storyboard ideas for your game:

Name of your Game:

Game Description:

Draw your game stage here:



Lessons Learned:

What was challenging? Where did you get stuck?

What did you learn? What was your A-ha! Moment?

What are some different types of games?

What features make a game interesting to you?

What features do you like and what code is used for it?

FEATURE	CODE

Lessons Learned:

What was challenging? Where did you get stuck?

What did you learn? What was your A-ha! Moment?