This document is been made to guide students and instructors during their discussions and hands-on activities. It is not intended to be complete by itself.

This document and related ones are available in digital form via www.cse.msu.edu/~stockman

It is better to print this document in color; however, if that is too expensive, black and white will do. Scratch uses color and the Scratch slides degrade in black and white. Experience has shown that this material is interesting and accessible to students from ages 10 and up. Younger students may have to take a little more time than those who are older.

Computer Science Concepts

1. **What is an algorithm?**
2. **Binary information coding**
3. **Programming concepts via Scratch**

Designed for learners 10 years old up to 100. **Coverage differs according to which camp.**
Materials for hands on activities

1) a set of cards with numbers, say, 42, 17, 43, 55, 20, 32, 12
2) A set of cards with binary place values: 1, 2, 4, 8, 16

3) Scratch downloaded from www.scratch.mit.edu
4) www.learnscratch.org for tutorials for homework

What is an algorithm?

What is an *algorithm*?

Algorithms for **average** and **min**

An algorithm for **sorting**

Algorithm for computing the average of two numbers

- Let the first number be A; say A=12
- And the second number be B; say B=15
- To compute the average __________
- So the average of 12 and 15 = ________
- WHAT IS THE AVERAGE ALGORITHM?
What is an algorithm?

An **algorithm** computes some output from the given input using precise steps that a machine can follow (machine or other agent).

**Hands on activity: finding MIN**

- Organize into 2-person teams
- Team has “person A” and “person B”
- Each team has a stack of cards
- There is one number on each card
- **No one looks at numbers yet**

**problem: finding min**

- Person B is the “worker” for A.
- Person A has cards, but cannot look at the numbers. A must find the min of all cards.
- B can look at only 2 numbers at a time -- B returns to A the smaller smaller first.
- **HOW CAN PERSON A FIND THE MIN?**
- Shuffle the cards
- A and B switch roles: new person A finds MIN
Thought Questions

- If A has 5 number cards, how many times does A call on B? _____
- If A has 10 cards, how many times does A call on B? _____
- If A has N cards, how many times does A call on B? _____
- Finding the MIN of a list of numbers has "complexity" ________

Problem: how to sort?

Use set of cards for each team. How can A sort using B as comparer?

TRY IT: instructor will check your final output

Sorting two-person activity

- This is a 2-person exercise. Person A CANNOT see the numbers – EVER!
- To solve the sorting puzzle, person A must form an ordered list of cards, which the instructor will check for order (this is the OUTPUT).
- Person B can only compare two numbers by seeing the numbers. Person B gives A the smaller number card first (precise operations).
- Person A can only ask Person B which of two cards has the smaller number. Person A never ever looks at any numbers.
- To solve the sorting puzzle, person A must start from an unordered list of cards (this is the INPUT).
Sorting thought questions

- How many times will A call on B to sort 5 numbers? _____
- How many calls to sort 10 numbers? _____
- How many calls to sort N numbers? _____

- There are many sorting algorithms. Some are fast when \( N \) is small but slow when \( N \) is big. Some are the opposite. Computer scientists learn these algorithms and many others.

Algorithms often do operations over and over

- Called “looping”
- Also called “iteration” or “repetition”
- “Recursion” is a related idea – first select the MIN from the list and then “recurse” on the rest of the list
Binary Information Coding

1. Only 2 communication symbols: WHY?
2. Can we code numbers, characters, music, images, movies, car designs using just 2 symbols? HOW?

Coding activities

1. Guessing the number of fingers in decimal
2. Guessing 1 finger versus 0 fingers
3. Encoding small integers using binary cards
4. Coding using animate versus inanimate objects ("My teacher plays rock, paper, scissors with me.")

* Instructor quickly brings, say, 7 fingers from behind the back. Students make errors in guessing the number. Few errors in 0 versus 1 discrimination.
15

- What number?
- What number?
- What number?

0 is 1 is

- What number? ______
- What number? ______
- What number? ______
- What number? ______

What number is coded?

red = 1; black = 0 ______
face = 0; non face = 1 ______
face = 1; non face = 0 ______
black = 1; red = 0 ______
Binary coding in a computer

- Symbol 0 can be represented by +5V and 1 by -5V
- 0 can be darkness and 1 lightness, as used on CDs and DVDs or in optical fibers

Birthday guessing game as time and local expert permits

- Member of audience writes day of month when born
- Instructor codes it using animate vs inanimate, or perhaps skin on skin vs skin on cloth
- Assistant in back of room guesses the day (12, 23, etc)

Guess Jo’s Age

My teacher plays rock, paper, scissors with me? ________

Think of a man, a woman, a ship, and an iceberg. ________

What’s the code?
Programming concepts via Scratch; Quick Intro

Hello!

GET YOUR OWN
Free download of software available at www.scratch.mit.edu
Video tutorials for learning available at www.learnscratch.org
Learn a lot on your own later.

What can we do with Scratch?

- Learn some computing concepts.
- Learn some practical algorithms.
- Use Scratch as computing tool.
- Have fun with Scratch creating stories, games, art. – it’s multimedia

Start scratch and let’s go!

- In the lab, Click on the cat icon
- Scratch programming environment comes up
- First do simple things
Explore Scratch environment

- **Stage** is at upper right (where actors act and information is displayed)
- **Sprite** is another name for actor.
- Instruction menus/panel at left (instructions for the actors)
- A **script** is another name for program or method; a script tells the actor what to do.
- Programming area in center; here is where we construct scripts for the sprites

The LOOKS menu

Has instructions for setting the color, size, and visibility of a sprite. Costumes will be used later.

The “hello” script (program)

- Choose the Looks menu
- Click on the “say hello” lego block
- Check your sprite behavior at the right
- Then click “say hello for 2 secs”

![say Hello!](image)

Your very first Scratch program!
Try some other looks operations *(click on menu items)*

- Change color effect by 25
- Change color effect by 25 again
- Hide
- Show
- Change size by 10
- Change size by 10 again
- Set size to 100%

Let’s write a script to

- Say “hello” for 2 seconds
- Then change color by 25
- Then think “Hmm..” for 4 seconds
- Then change color by 75
- Then change size by 200

**Drag each instruction from the menu to the center script area.** Connect them into a single block. Edit the parameters to get the numbers we want.
Our script (program)

A sequence or block is a simple script or *program*

- The first instruction is done first
- The second instruction is done second
- The last instruction is done last.
- (if any one instruction is done, then every one of them is done)

The WAIT instruction

- WAIT is needed to slow down the acting so we can see or hear it properly (computers are too fast sometimes)
- Get the wait instruction from the CONTROL menu. Ignore the other menu options for now.
- Insert a wait in our looks script
3 second pause between changing color and size

Student exercise: script your sprite to do ...

- Double the size of the sprite
- Wait 2 seconds
- Change the color of the sprite to green
- Wait 4 seconds
- Change the whirl effect to 100
- Say “My nose is tired!”
The MOTION menu

How to locate and orient a sprite; and, how to move a sprite.

Position on the stage

- Using the **Looks menu**, shrink our cat to 25%.
- Click on the **Motion menu**.
- Click to check the box for x-position and y-position

Drag your cat around and note its x-y position.

Exercises: goto instruction (do you know about coordinates?)

- In the Motion menu, drag the “goto xy” instruction to the script panel.
- Edit the **coordinates** and click to see the sprite’s position
  - A) goto x=200, y=0
  - B) goto x=-200, y=0
  - C) goto x=200, y=-100

Where does the cat go?
The “glide-to” instruction

- Drag the “glide-to” instruction into your script panel.
- Edit the coordinate values and click to see where your sprites goes.

Exercise

- Create a script to glide the sprite along the sides of a triangle. The first vertex of the triangle is (-100, -100). The second vertex is (200, -100). The third vertex is (50, 100). Make sure you complete the triangle.
- Change the speed of gliding and run again.
- New feature: click on Pen Down in the Pen Menu and run it again
Tracing a square: 4 sides

REPEAT move and turn 4x
Detroit Tiger runs the bases

- Import the baseball park background
- Put your tiger at home plate (how?)
- Have it glide to first base
- Have it glide to 2nd base
- Then 3rd and home
- Make it do a flip
- Make it say “Home run!”

Bat flying example

- Bat seems to move randomly
- 2 costumes create motion
- Press keys to control program
- Left arrow moves bat left
- Right arrow moves bat right
Exercise: bat flying randomly

Make a bat fly by flapping between 2 bat “costumes”. Use random values of coordinates x and y.
Use arrows keys to control the bat’s position.
* Use a forever loop to make the bat continue to fly when the flag is clicked
* See if you can create a “score” variable and use it to count how many times the user clicks on the bat.

Adding new features

- Create a score variable.
- Increase score by 2 when user clicks on the bat.
- Decrease the score by 1 when the user misses clicking on the bat.
- Add a bat sound to the game.
- Change the sound when the user clicks on the bat.
Learn from others: Scratch Library of Examples

- Click on “File”
- Click on “Open”
- Click on “Examples”
- Then “Stories”
- Then Madlib
- Run it; study the code; change the code

Do the story and study the scripts

Study other examples. Change the scripts to change them.
Exercise: find the MIN of a list

- Explore how to create a list variable
- Explore how to input numbers to the list
- Explore how to pass through the list to set a variable MIN to the minimum in the list
- Remember: www.learnscratch.org can help your learning

Doing your own exercises is important. Use www.learnsractch.org and www.scratch.mit/edu for help.

- Try different variations on the themes.
- Learn how to use a “background” picture on your stage.
- Learn how to make sounds or music in your loops.
- Learn how to detect when two of your sprites collide.
- And so forth and so on for infinite fun.