AP: Discuss with a shoulder partner these questions.

EQ Question:

1. What does the computer understand?
2. How is information converted to machine language?
3. What is binary?
4. What is hexadecimal?
5. What is octal?

Digital Information

- A computer stores information digitally as binary numbers.
    - numbers
    - text
    - graphics and images
    - video
    - audio
    - program instructions

- In some way, all information is *digitized* - broken down into pieces and represented as numbers

Binary Numbers

- Binary number system has only two digits

  0 and 1

- A single binary digit (0 or 1) is called a *bit*
- A single bit can represent two possible states, like a light bulb that is either on (1) or off (0)
- Binary is Base 2 number system
- so there are 2^N permutations of N bits
- It takes 8 bits to make a byte or a character that represents a form of data.

Ascii Code

- http://sticksandstones.kstrom.com/appen.html
- http://www.ascii.cl/conversion.htm

Bit Permutations

Therefore, N bits are needed to represent 2^N unique items

How many items can be represented by

1 bit ?	$2^1 = 2$ **items**
2 bits ?	$2^2 = 4$ **items**
3 bits ?	$2^3 = 8$ **items**
4 bits ?	$2^4 = 16$ **items**
5 bits ?	$2^5 = 32$ **items**
6 bits ?	$2^6 = 64$ **items**
7 bits ?	$2^7 = 128$ **items**
8 bits ?	$2^8 = 256$ **items**

Bit Permutations

Each additional bit doubles the number of possible permutations

1 bit 2 items	2 bits 4 items	3 bits 8 items	4 bits 16 items	
0	00	000	0000	1000
1	01	001	0001	1001
	10	010	0010	1010
	11	011	0011	1011
		100	0100	1100
		101	0101	1101
		110	0110	1110
		111	0111	1111

Question: 3 bits can represent how much data? 2^3

8

What is Binary

The word **binary** means two. The binary number system has two symbols: 0 and 1. When we write binary numbers we use a "2" for a *subscript* to represent the binary system.

A "**bit**" (short for "*binary digit*") is the smallest piece of data that a computer knows. It is a single digit, which can be a one or a zero. A "**word**" is a group of any number of bits. A "**byte**" is a group of 8 bits, You would have 256 different combinations if you wrote down all the different possible combinations of ones and zeros that could make up a byte.

One – on

Zero – off

Add up the place value for everyone 1.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
128	**64**	**32**	**16**	**8**	**4**	**2**	**1**	**Answer**
1	0	1	0	1	0	0	1	
0	0	1	1	0	0	1	0	
0	0	1	1	1	0	0	0	
0	1	1	0	0	0	1	0	
1	1	1	0	1	1	1	0	
1	1	1	0	0	0	0	1	

Binary to Decimal

Mathematical Equation to convert from Binary to Decimal

Binary to Decimal Find the place value for everyone. Multiple 1 by that place value.

1	0	1	0	1	0	0	1
2^7	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**

1×2^7 1×2^5 1×2^3 1×2^0 =
128 + 32 + 8 + 1 **= 169**

Converter Tool with explanation. http://acc6.its.brooklyn.cuny.edu/~gurwitz/core5/nav2tool.html
Cisco Binary Game:

Practice:

Convert 1100011 to decimal

From Binary to Hexadecimal

Hexidecimal, or base 16, number system is a common system used with computers. It is a human-friendly representation of binary-coded values in computing and digital electronics. It is used in web pages for colors.

Hexadecimal is represented by 16 digits 0-9 and then A – F

A = 10, B = 11, C = 12, D = 13, E = 14, F = 15

Converting Binary to Hex

10001100101001

STEP ONE: Take the binary number and from right to left, group all placeholders in groups of 4. Add leading zeros, if necessary:

$\mathbf{0010 \quad 0011 \quad 0010 \quad 1001_2}$

STEP TWO: Convert each triplet to its single-digit octal equivalent. (**HINT:** For each group of 4, the hex conversion is the same as converting to a decimal number):

$\mathbf{0010 \quad 0011 \quad 0010 \quad 1001_2}$

2 3 2 9_{hex}

$\mathbf{2329_{16}}$

You do it: Practice: convert 11001110 to Hex

Convert from Hex to Binary

Converting Hex to Binary

$\mathbf{2329}_{16}$

STEP ONE: Take each hex digit and convert each digit to a binary form. Keep leading zeros:

0010 0011 0010 1001

$\mathbf{2329}_{16} = 10001100101001$

You do it: Practice: convert 4AB to Binary

Octal - Base 8 Numbering System

The octal, or base 8, number system is a common system used with computers. Because of its relationship with the binary system, it is useful in programming some types of computers.

Converting Binary to Octal

Take the binary number 10001100101001

STEP ONE: Take the binary number and from right to left, group all placeholders in group of 3. Add leading zero at end if necessary:

010 001 100 101 001_2

STEP TWO: Find the binary number for that group of 3.

010 001 100 101 001_2

2 1 4 5 1 oct

10001100101001 = **21451_8**

You do it: Practice: convert 11001110 to Oct

Converting Octal to Binary

43520 $_8$

Bring the numbers the binary group of 3

4	3	5	2	0
100	011	101	010	000

100011101010000 = 43520 oct

You do it: Practice: convert 2643 to binary

Convert any base to decimal by multiplying

Decimal	Base 10	89	$(8 \times 10^1) + (9 \times 10^0)$ = 89
Binary	Base 2	1011001	$(1 * 2^{6)} + (1 * 2^4) + (1 * 2^3) + (1 * 2^0)$ = 89
Hexadecimal	Base 16	59	$(5 * 16^1) + (9 * 16^0)$ = 89
Octal	Base 8	131	$(1 * 8^2) + (3 * 8^1 + (1 * 8^0)$ = 89

programming and the other with none or little)

Go to my website and start on the SNAP programming exercises. Go through the powerpoints and do each of the exercises listed.

SNAP Website

PPT for SNAP Project

All programs have 3 basic control structures: Sequential, Condtional, Iteration (loop)

SNAP Exercises #1 - Moving and Talking (see ppt slide #8)
SNAP Exercises #2 - Squares (see ppt slide #10)
SNAP Exercises #3 - Triggers (see ppt slide #12)
SNAP Exercises #4 - Threads (see ppt slide #14)
SNAP Exercises #5 - Loops (see ppt slide #20)
SNAP Exercises #6 - Input (see ppt slide #24)
SNAP Exercises #7 - Arithmetic (see ppt slide #26)
SNAP Exercises #8 - Conditionals / Boolean (see ppt slide #32)
SNAP Exercises #9 - Events - MarcoPolo (see ppt slide #34)
SNAP Project - Turtles