Binary Basics

Provided by TryEngineering - www.tryengineering.org

Lesson Focus

This lesson is intended to provide very young students with a basic understanding of how the system of binary numbers works.

Lesson Synopsis

The lesson begins by asking how a Stone Age man managed to go home to his cave and tell his family he had caught 11 fish when he only has 10 fingers? It then draws students' attention to the curious fact that whereas, when we write a word on paper or on a blackboard, we begin at the left and gradually work over to the right. But when we write numbers we start over on the right, and work over to the left. And so it is the same with binary numbers. We then explain how simple binary numbers really are.

The lesson notes the convenient similarity between the binary system, which uses only two digits, a **0** or a **1** for everything it does, and electronic computers, which essentially asks itself two questions, **No** or **YES.** In essence, the binary system and computers seem to have been made for each other. There is also a side reference to the Morse Code, which also employs two symbols – a **Dot** and a **Dash**.

This lesson has been assembled with the youngest students in mind. It provides the binary version of all 26 letters in the alphabet, so that students can send coded messages to each other. The lesson ends with a section in which the students are invited to discuss with the teacher, various ways in which they think these demonstrations could be improved.

Age Levels

9 – 12

Objectives

Students will:

- Gain a useful initial acquaintance with the basis of binary numbers, and by inference how computers work.
- Learn about the importance of discipline and team work.

Anticipated Learner Outcomes

Students will demonstrate/explain how:

- the Binary System works and its similarity to today's electronic computers.

Resources/Materials

- Teacher Resource Documents (PPT Slides).
- Student Resource Sheets (PPT Slides)

Alignment to Curriculum Frameworks

- See attached curriculum alignment sheet.

Internet Connections

- TryEngineering (www.tryengineering.org)
- ITEA Standards for Technological Literacy: Content for the Study of Technology (www.iteaconnect.org/TAA)
- NSTA National Science Education Standards (www.nsta.org/publications/nses.aspx)

Optional Writing Activity

- Having discussed the obvious limitations of this very simple demonstration, students should be asked to set out ways in which they think it could be improved.

Binary Basics

For Teachers:
Teacher Resources

◆ **Objectives**

Students will:

- ✦ Gain a useful initial acquaintance with the basis of binary numbers, and by inference how computers work.
- ✦ Learn about the importance of discipline and team work

◆**Materials**

- ✦ For all intents and purposes, the material costs for the demonstrations in this lesson are zero; ten sheets of paper only, as set out in the hands-on section set out at the end of this lesson.

◆ **Time Needed**

- ✦ It is suggested that, for the younger students, aged between 8 and 10, three sessions of 45 minutes each should be sufficient. For older students, two such sessions should be enough.

◆ **Procedure**

A Quandary

Who knows what a quandary is? Does your neighbor know?

A quandary was what a distant early cousin of the Flintstones was in when he caught 11 fish. He was really pleased, but on the way home he began to wonder how he could explain eleven fish when he didn't have enough fingers.

Roman Numbers

Despite all the wonderful things the Romans did, they were really out to lunch when it came to even simple arithmetic.

They had no numbers of any sort and had to use letters.

THIS IS ALL THEY HAD

1	**I**	**50**	**L**
2	**II**	**100**	**C**
3	**III**	**500**	**D**
4	**IV**	**1000**	**M**
5	**V**		
6	**VI**		
7	**VII**		
8	**VIII**		
9	**IX**		
10	**X**		

For example, this year's date (2016) comes out as:-

MMXVI, which is: 1000+1000+10+5+1

How do you multiply:- XI by C?

Even an abacus can’t do that.

And if you are still having difficulty, go have a look at your town clock. Chances are it has Roman Numerals.

P.S. There are charts on the Net which help you convert from Roman to our numbers which actually are Arabic.

A Little Background
Yes, the Romans built the Coliseum and wonderful temples. But only after much trial and error.

Building everyday buildings more than about 3 floors was a challenge because there was no easy way to do simple sums.

Roman history books are full of accounts of ordinary houses falling down.

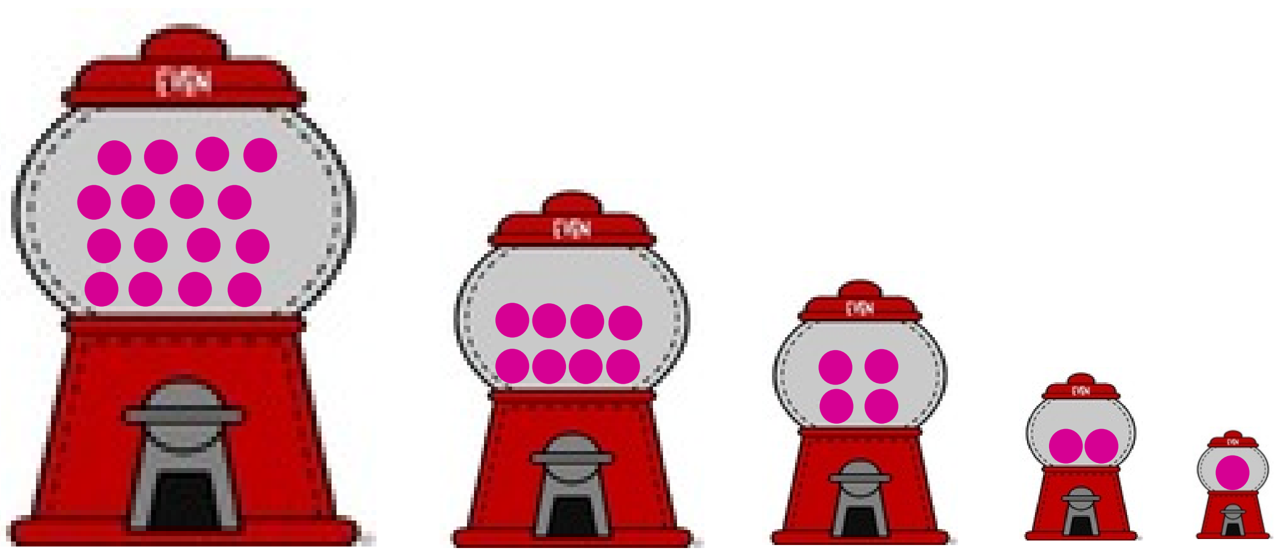
So today we write all our letters starting on the left and moving to the right. But have you noticed that when we write numbers, we start on the RIGHT, and as they get bigger, we move over to the LEFT.

Notice: **1**
10
100
1000
10000

Binary numbers work the same way.

WHAT DO YOU SEE HERE?

NOW VISUALIZE A ROW OF 5 GUMBALL MACHINES



- The one on the far right is very small and can only hold 1 gumball (max)
- The next one to the left is twice as big and can hold 2 gumballs (max)
- Once more to the left is a machine twice as big again. It can hold 4 gumballs.
- After that comes a machine capable of holding twice as many again, or 8 gumballs.
- Finally, the last machine in our line is a jumbo that can hold twice as many as the one before it, or 16 gumballs.

Each of these machines has a flap at the bottom, so that when you pull the lever, ALL the gumballs in that machine fall out.

In other words, these machines are always either completely full, or completely empty.

Now, you might be wondering “If gravity is emptying these machines – how are they filling back up exactly?”

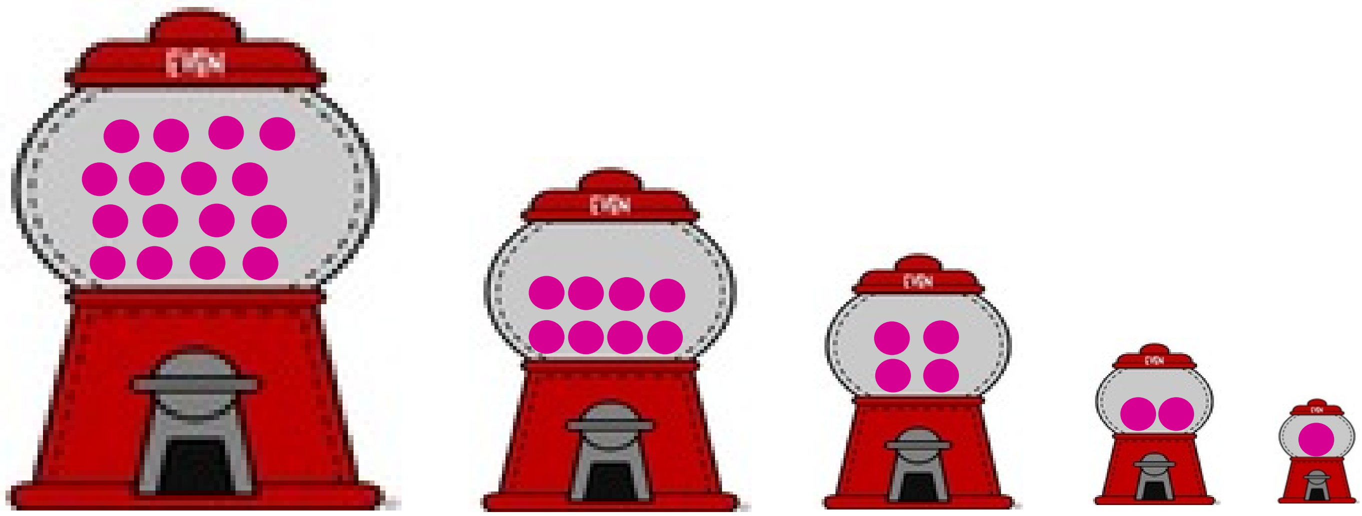
This is not a problem, because there is a small person behind the scenes happy to refill the machines for us - but only when told to do so!

So What Do We Have?

A row of five machines capable of holding gumballs as follows:

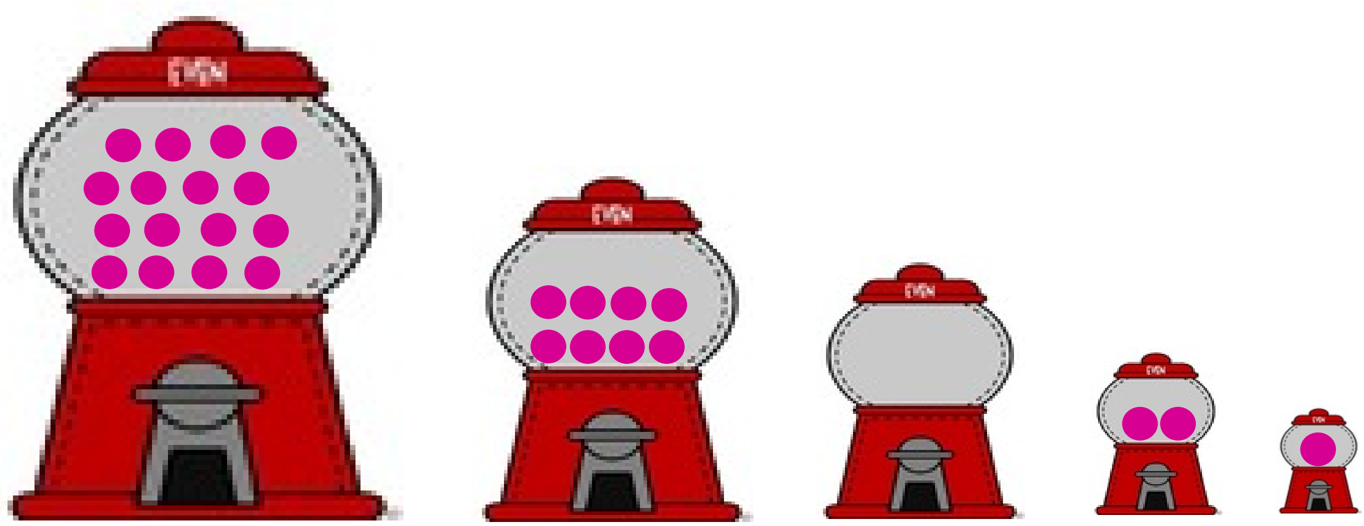
16 8 4 2 1

Now assuming all our machines are full, how many gumballs do we have?



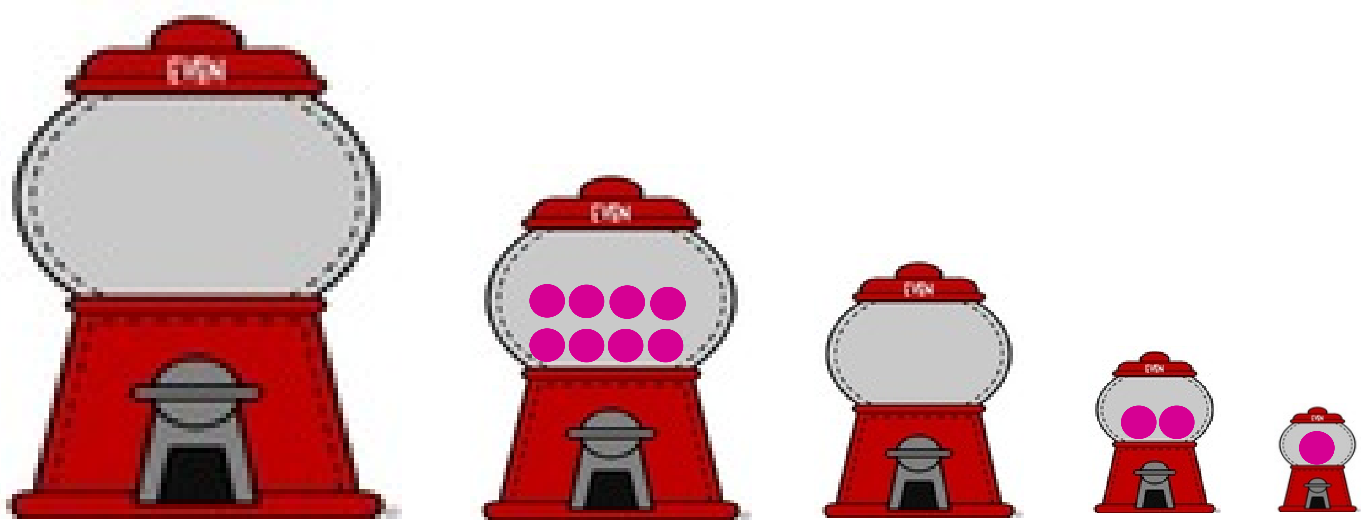
16 + 8 + 4 + 2 + 1 = 31

Assume the middle machine is empty; how many do we have?



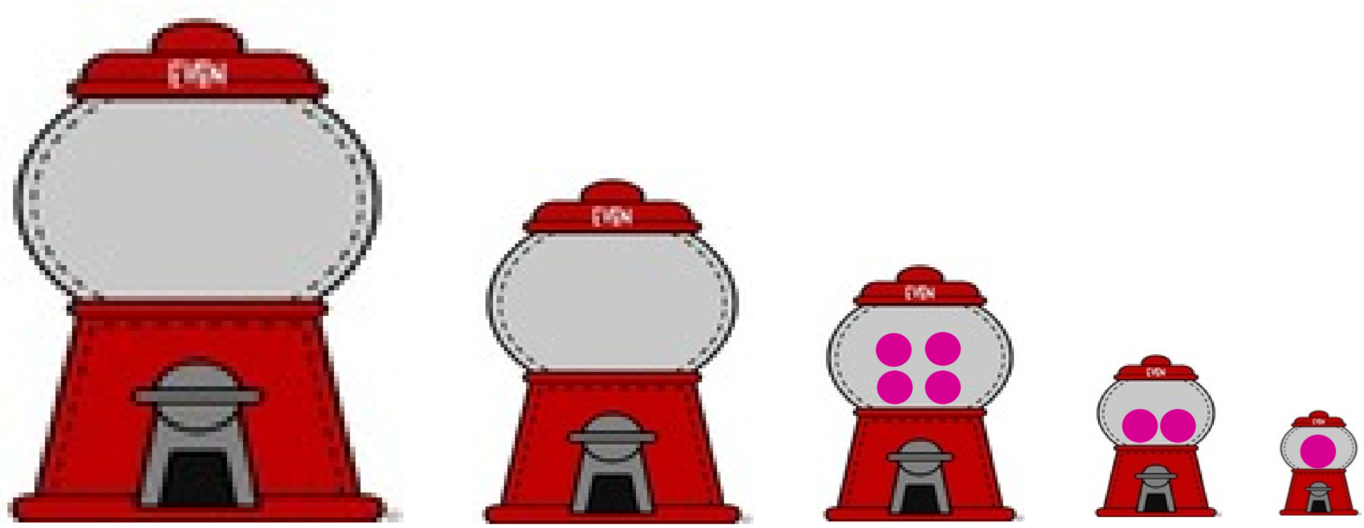
16 + 8 + 0 + 2 + 1 = 27

Assume the left-hand machine and the middle one are empty; what do we have?



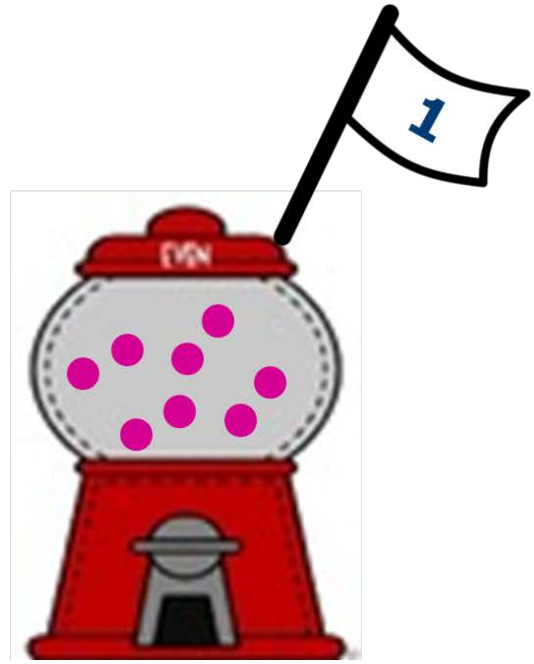
0 + 8 + 0 + 2 + 1 = 11

Assume only the first two machines are empty, we would have



0 + 0 + 4 + 2 + 1 = 7

It would help our small person if each of our gumball machines had two flags.



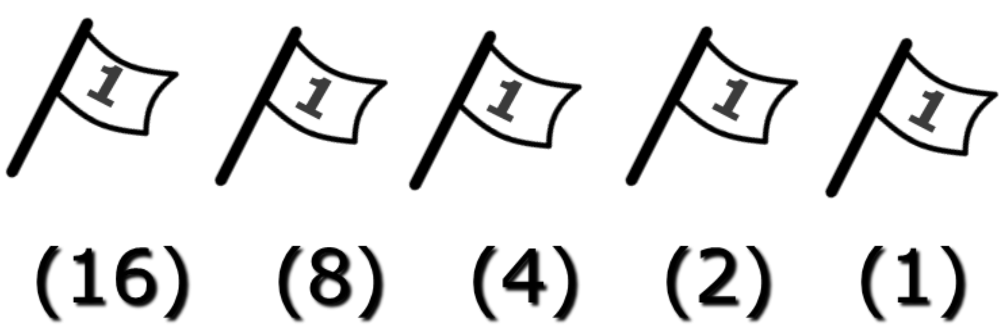
HOW MIGHT THESE FLAGS HELP OUR SMALL PERSON?

Turn and share your thoughts with a partner. Why use the flags? Do you agree?

NOW WHAT DO WE HAVE?

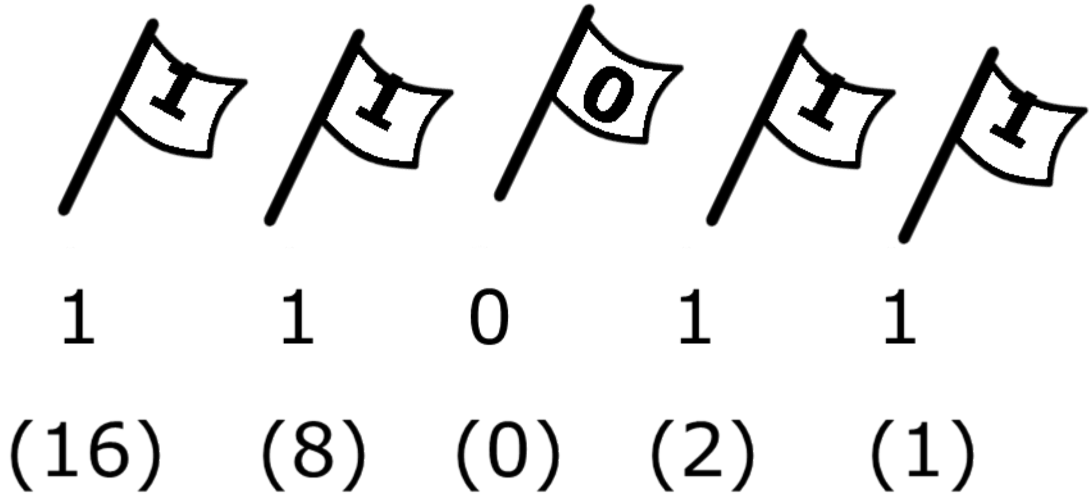
Think about our row of 5 gumball machines, each of which has a "0" flag or a "1" flag

If all machines are full, we would see five flags showing a "1" like this:



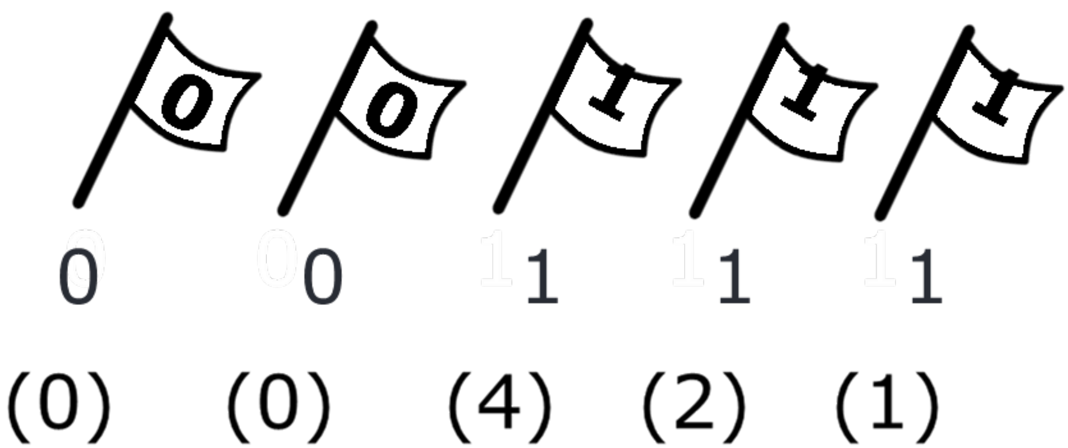
Which means we have 31 gumballs

If the middle machine is empty, we would see:



Which means we have 27 gumballs

And if the two machines on the left are empty we would see:



How many gumballs do we have now???

Explain how you know to a partner.

Binary Chairs

1. Take 5 chairs and line them up.
2. Take 5 sheets of 8-1/2 x 11 paper.
3. Take a felt pen and write a “1’, a “2”, a “4” an “8” and a “16” on them (one on each sheet).
4. Stick the sheet with “1” on it, on the back of the chair on the extreme **RIGHT**.

5. Then moving over to **LEFT**, stick the sheets with “2”, “4”, “8” and “16” on the corresponding chairs.
6. In other words, when you have finished, you will have 16, 8, 4, 2, and 1 on the backs of the chairs.
7. Take 5 students at random and seat one student on each of the chairs.
8. Take 5 more sheets of paper and on one side of **EACH** sheet mark a “1”.
9. Then turn **each** sheet over and put a “0” on **EACH** sheet.
10. Give one of these sheets of paper to each of the 5 students.
11. Sit all the other students facing the five mentioned above.
12. Have each of the 5 selected students hold up their piece of paper in any order they like.

Suppose they come up looking like this.

Wall		16		8		4		2		1		
The 5 seated students		1		0		0		1		0		
Total as seen by the class	**=**	**16**	**+**	**0**	**+**	**0**	**+**	**2**	**+**	**0**	**=**	**18.**

Then have the 5 seated students rearrange their papers in some other order, such as:

Wall	16		8		4		2		1	
The 5 seated students	0		1		1		0		1	
Total as seen by the class	**0**	**+**	**8**	**+**	**4**	**+**	**0**	**+**	**1**	**= 13**.

How does a computer use binary numbers to make the letters of our alphabet?

Turn and talk to a partner about your prediction.

They use a system of 26 coded numbers.

CODED LETTERS

A	=	**010**00001	J	=	**010**01010	S	=	**010**10011
B	=	**010**00010	K	=	**010**01011	T	=	**010**10100
C	=	**010**00011	L	=	**010**01100	U	=	**010**10101
D	=	**010**00100	M	=	**010**01101	V	=	**010**10110
E	=	**010**00101	N	=	**010**01110	W	=	**010**10111
F	=	**010**00110	O	=	**010**01111	X	=	**010**11000
G	=	**010**00111	P	=	**010**10000	Y	=	**010**11001
H	=	**010**01000	Q	=	**010**10001	Z	=	**010**11010
I	=	**010**01001	R	=	**010**10010			

Note 1: The leading "010" above indicates an upper case letter. It is shown in bold here simply for ease of identification.

Note 2: A leading "011" would indicate a lower case letter.

Note 3: For example, "C", which is No. 3 in the alphabet is also 3 in binary. "Z", which is No. 26 in the alphabet, is also 26 in binary terms, which figures.

NOW TRANSLATE THE FOLLOWING MESSAGE, USING THE TABLE ON YOUR HANDOUT
01000001

01001001
01010011

01000110
01001111
01010010

01000001
01010000
01010000
01001100
01000101

SO NOW YOU CAN SEND CODED MESSAGES TO YOUR FRIENDS

01001000, 01000001, 01010110, 01000101

01000001

01001110, 01001001, 0100011, 01000101

01000100, 01000001, 01011001 !

TRY IT!

TURN YOUR HANDOUT OVER AND WRITE A MESSAGE TO A FRIEND IN BINARY CODE (use a complete sentence with several words) ***Consider using a number as well (smaller than 100 please)

WHEN WE'RE FINISHED WE'LL TRADE AND DECODE!!!

How many numbers can a computer store?

- Binary Digit = 1 "Bit"
- 8 Bits = 1 "Byte"
- 1000 Bytes = 1 "Kilobyte"
- 1000 Kilobytes = 1 "Megabyte"
- 1000 Megabytes = 1 "Gigabyte"
- Typical home computer in 2015 → 1000 Gigabytes = 1 "Terabyte"
- 1000 Terabytes = 1 Petabyte
- 1000 Petabytes = 1 Exabyte
- 1000 Exabytes = 1 Zettabyte
- 1000 Zettabytes = 1 Yottabyte
- 1000 Yottabytes = 1 Brontobyte
- 1000 Brontobytes = 1 Geopbyte

WHY ARE BINARY NUMBERS SO USEFUL?

Despite what you might think, computers are really not smart. All they do is look for zeros and ones, time after time after time.

And of course, computers never get tired, never take coffee breaks, and (usually) never make mistakes.

The smart part comes with the people who write the "Computer Code" which organizes the zeros and ones into something meaningful.

AN HISTORICAL NOTE

Who has not heard of the "Morse Code"?

- Although not a binary system, there are certain similarities.
- It was developed by Samuel Morse in 1845, as a system of sending messages electrically over long distances by wire.
- This was found to be of great use to the railroads, and in 1866 for sending messages across the Atlantic via submarine cable. Later still it was used by Marconi (1901) for sending messages by radio. In some cases it is still in use today.
- To each of the 26 letters of the alphabet, Morse allocated a group of three electric pulses, some short and others long, formed by the operator holding down his "key" for long or short periods of time.

- The most famous Morse Code group of letters is S O S, which amongst ships in distress stands for “Save our Souls”.

  S is dot dot dot.
  O is dash dash dash.
  S is dot dot dot.
  Or … _ _ _ … Easy to remember and easy to do.

- An early application was when the Titanic sank in 1912.
- The full Morse Code is available on the net.
- The first message sent by Morse in 1845 was “What Hath God Wrought”.
- He was also a famous portrait painter, which shows you don’t have to be a specialist.

Name: ____________________ Date: ________________

BINARY CODED LETTERS

A	=	01000001	J	=	01001010	S	=	01010011
B	=	01000010	K	=	01001011	T	=	01010100
C	=	01000011	L	=	01001100	U	=	01010101
D	=	01000100	M	=	01001101	V	=	01010110
E	=	01000101	N	=	01001110	W	=	01010111
F	=	01000110	O	=	01001111	X	=	01011000
G	=	01000111	P	=	01010000	Y	=	01011001
H	=	01001000	Q	=	01010001	Z	=	01011010
I	=	01001001	R	=	01010010			

Binary Basics

For Teachers:
Alignment to Curriculum Frameworks

Note: Lesson plans in this series are aligned to one or more of the following sets of standards:

- U.S. Science Education Standards (http://www.nap.edu/catalog.php?record_id=4962)
- U.S. Next Generation Science Standards (http://www.nextgenscience.org/)
- International Technology Education Association's Standards for Technological Literacy (http://www.iteea.org/TAA/PDFs/xstnd.pdf)
- U.S. National Council of Teachers of Mathematics' Principles and Standards for School Mathematics (http://www.nctm.org/standards/content.aspx?id=16909)
- U.S. Common Core State Standards for Mathematics (http://www.corestandards.org/Math)
- Computer Science Teachers Association K-12 Computer Science Standards (http://csta.acm.org/Curriculum/sub/K12Standards.html)

◆National Science Education Standards Grades K-4 (ages 4 - 9)

CONTENT STANDARD E: Science and Technology

As a result of activities, all students should develop

- ✦ Understanding about science and technology

◆Principles and Standards for School Mathematics

Number and Operations Standard

As a result of activities, all students should develop

- ✦ Understand numbers, ways of representing numbers, relationships among numbers, and number systems.
- ✦ Compute fluently and make reasonable estimates.

Connections Standard

As a result of activities, all students should develop

- ✦ Recognize and apply mathematics in contexts outside of mathematics.

◆Standards for Technological Literacy - All Ages

The Nature of Technology

- ✦ Standard 3: Students will develop an understanding of the relationships among technologies and the connections between technology and other fields of study.

Technology and Society

- ✦ Standard 7: Students will develop an understanding of the influence of technology on history.

◆CSTA K-12 Computer Science Standards Grades K-3 (ages 5-8)

5.1 Level 1: Computer Science and Me (L1)

- ✦ Computational Thinking (CT)
  1. Use technology resources (e.g., puzzles, logical thinking programs) to solve age-appropriate problems.
  5. Demonstrate how 0s and 1s can be used to represent information.
- ✦ Collaboration (CL)
  2. Work cooperatively and collaboratively with peers, teachers, and others using technology.
- ✦ Computer Practice and Programming (CPP)

◆CSTA K-12 Computer Science Standards Grades 3-6 (ages 8-11)

5.1 Level 1: Computer Science and Me (L1)

- ✦ Computational Thinking (CT)
  3. Demonstrate how a string of bits can be used to represent alphanumeric information.