

Activity 1

Count the dots—*Binary numbers*

Age group Early elementary and up.

Abilities assumed Counting up to 15 or 31, matching, sequencing.

Time 10 to 40 minutes.

Size of group From individuals to the whole class.

Focus

Representing numbers in base two.

Patterns and relationships in powers of two.

Summary

All data in a modern digital computer is ultimately stored and transmitted as a series of zeros and ones. This activity demonstrates how numbers and text can be represented using just these two symbols.

Technical terms

Binary number representation; binary to decimal conversion; bits and bytes; character sets.

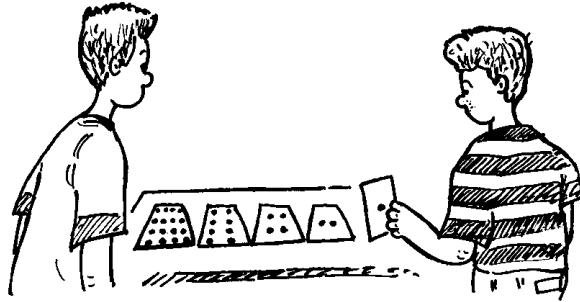


Figure 1.1: Initial layout of the binary cards

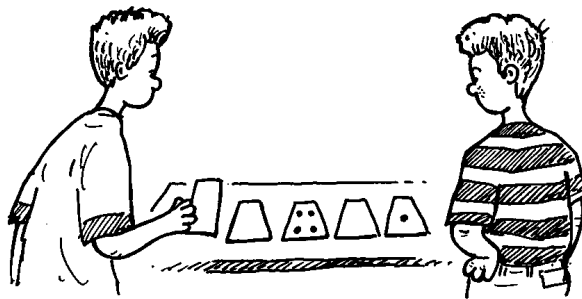


Figure 1.2: Flipping the cards to show five dots

Materials

Each child will need:

- one set of five cards from the blackline master on page 17 (the blackline master has two sets),
- a copy of the blackline master on page 18, and
- a pen or pencil.

What to do

1. Seat the children where they can see you, and give each child a set of cards.
2. The children should lay their cards out, as in Figure 1.1, with the 16-dot card to their left. Some children will be tempted to put the cards in the opposite order, so you should check that they are in descending numeric order from left to right. For younger children, do not use the 16-dot card.

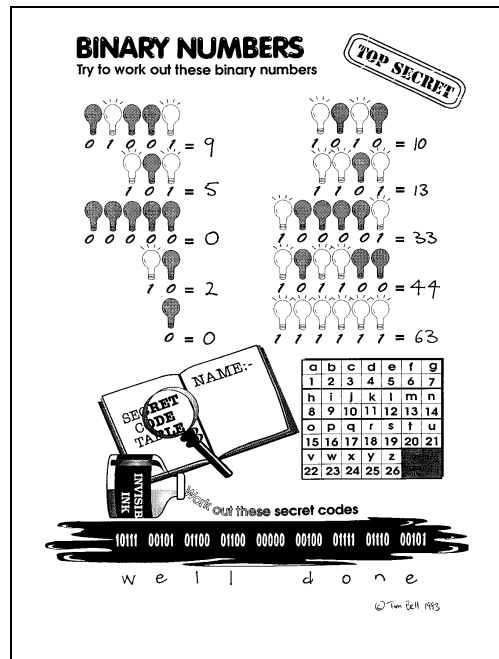


Figure 1.3: Solution to the worksheet on page 18

- Have the children work out which cards to flip over so that exactly five dots are showing. The only (correct) way to do this is to have the 4-dot and 1-dot cards face up, and the rest face down (Figure 1.2). Each card must be either face up or face down, with all or none of its dots showing. Be prepared for some novel ways of getting five dots—it is not unusual for children to produce the requisite number by using spare cards to cover up three of the dots on the eight card!
- Now get the children to show other numbers of dots, so that they explore which numbers can be represented.

Ask for numbers such as three (requires cards 2 and 1), twelve (8 and 4), nineteen (16, 2 and 1) and so on. For those who find the combination for a number quickly, ask if they can find another way to get the number (there is only one way to display each number, and they are likely to discover this eventually).

Discuss what the biggest number is that can be made with the cards (it is 31 for five cards, 15 for four cards). The smallest? (Often the number one will be offered first, but the correct answer is zero.) Is there any number between the smallest and largest that can't be represented? (No—all numbers can be represented, and each has a unique representation.)
- For older children, ask them to display the numbers 1, 2, 3, 4, ... in sequence, and see if they can work out a procedure for incrementing the number of dots displayed on the cards by one (the number of dots increases by one if you flip all cards from right to left until you turn one face up).

6. This part of the activity uses zeros and ones to represent whether a card is face up or not.

Tell the children that we will use a 0 to show that a card is hidden, and a 1 if its face is showing. For example, the pattern in Figure 1.2 is represented by 00101. Give them some other numbers to work out (e.g. 10101 represents 21, 11111 represents 31). With some practice the children will be able to convert in both directions. You could ask children to take turns calling out the day of the month that they were born on using zeros and ones, and have the rest of the class interpret the date.

This representation is called the binary system, also known as base two.

7. Use the worksheet on page 18 to extend the exercise. (A completed worksheet is shown in Figure 1.3.)

The worksheet uses a light bulb that is switched on to represent a card that is showing, and a light bulb that is off to represent a hidden card. The first few patterns should be easy to work out. For example, the first pattern has the 8 and 1 cards showing, so the value represented is $8 + 1 = 9$. For the patterns with fewer than five light bulbs, the children should use only the smaller valued cards. For example, the second pattern has only three light bulbs, which correspond (from left to right) to the 4-, 2-, and 1-dot cards respectively. See if the children can work this out for themselves.

The six-bulb questions are designed to make the children think about how many dots should be on a sixth card. The number of dots on each card is double the number on the previous one, so the sequence is 1, 2, 4, 8, 16, 32, 64 Thus a 32-dot card would be added (at the left) to solve a problem that needs six cards.

The code at the bottom of the worksheet uses the numbers 1 to 26 to represent the letters of the alphabet. (A zero can be used to represent a space.) The children must work out what each number in the code is, and look up the corresponding letter in the table. This shows how a textual message can be converted to a series of zeros and ones. The children can then write coded messages for each other.

Variations and extensions

Instead of using cards with dots, the exercise can be done with the lengths of rods (a set of rods of lengths 1, 2, 4, 8 and 16 units can be used to create any length from 0 to 31) or with weights (a set of weights of 1, 2, 4, 8 and 16 units can be combined to produce any weight from 0 to 31).

Instead of calling out sequences like 01101, try using beeps—call out a high-pitched beep for a one and a low-pitched one for a zero. This activity is noisy in the classroom, but children find it memorable! Modems and fax machines use tones like this to transmit information, although the tones are sent so quickly that they blend to make a continuous screeching sound. If the children aren't familiar with this, they could try calling a fax machine number to hear what it sounds like.

Any objects that have two states can be used to represent numbers. Figure 1.4 shows some different ways of representing the number nine (01001). A particularly challenging method is to use your fingers. If a finger is up it represents a one; down represents zero. Counting on your

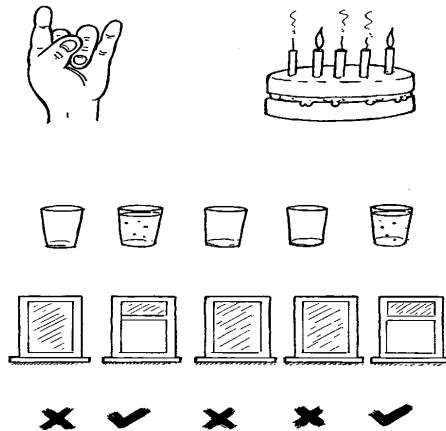


Figure 1.4: Some unusual ways of representing the number nine (01001 in binary)

fingers using the binary system enables you to go up to 31 on one hand, and 1023 on two hands. It requires some dexterity, and you have to watch out for rude gestures along the way! For a real challenge, try using your toes as well—this will allow you to count up to more than a million. (How many exactly? Two hands give 1024 possibilities, 0 through 1023. Hands and toes give $1024 \times 1024 = 1,048,576$ possibilities, 0 through 1,048,575.)

Older children will enjoy extending the sequence 1, 2, 4, 8, 16, 32 . . . The sequence contains an interesting relationship: if you add the numbers from the beginning from left to right, the sum will always be one less than the next number in the sequence.

Another property of binary numbers is that you can double the number by inserting a zero on the right-hand side of a number. For example, 1001 (9) doubled is 10010 (18). Older children should be able to explain why this happens. (All of the places containing a one are now worth twice their previous value, and so the total number doubles. The same effect occurs in base ten, where inserting a zero on the right of a number multiplies it by ten.)

Binary numbers are closely related to the guessing game in which one person thinks of a number and someone else tries to guess it by asking questions of the form “is it greater than or equal to x ?” For example, suppose the number is known to be less than 32. A sensible first question would be “is it less than 16?” The yes/no answers to the questions are given by the zero/one bits in the binary representation of the number. This is explored in detail in Activity 5.

The five-bit code used for letters does not allow both upper- and lower-case letters to be represented. You could have the children work out how many different characters a computer has to represent (including digits, punctuation, and special symbols such as \$), and consequently how many bits are needed to store a character. (With two lots of 26 letters, 10 digits, and a few punctuation marks, there are bound to be more than 64 codes needed, so at least seven bits are necessary. Seven bits allows for 128 characters, and this is more than sufficient.) Most current computers use a representation called ASCII (American Standard Code for Information Interchange), which is based on using seven bits per character. Longer codes that allow for the languages of non-English speaking countries are now becoming common.

What's it all about?

Modern digital computers almost exclusively use the system described above to represent information. The system is called binary because only two different digits are used. It is also known as base two (as opposed to base ten, which humans normally use). Each zero or one is called a “bit”, the term being a contraction of *binary digit*. A bit is usually represented in a computer's main memory by a transistor that is switched on or off, or a capacitor that is charged or discharged. On magnetic disks (floppy disks and hard disks), bits are represented by the direction of a magnetic field on a coated surface, either North-South or South-North. CD-ROMs store bits optically—the part of the surface corresponding to a bit either does or does not reflect light. When data must be transmitted over a telephone line or radio link, the ones and zeros are commonly represented by high- and low-pitched tones.

One bit on its own can't represent much, so they are usually grouped together as in the exercise above. It is very common to store bits in groups of eight, which can represent numbers from 0 to 255. A group of eight bits is called a byte.

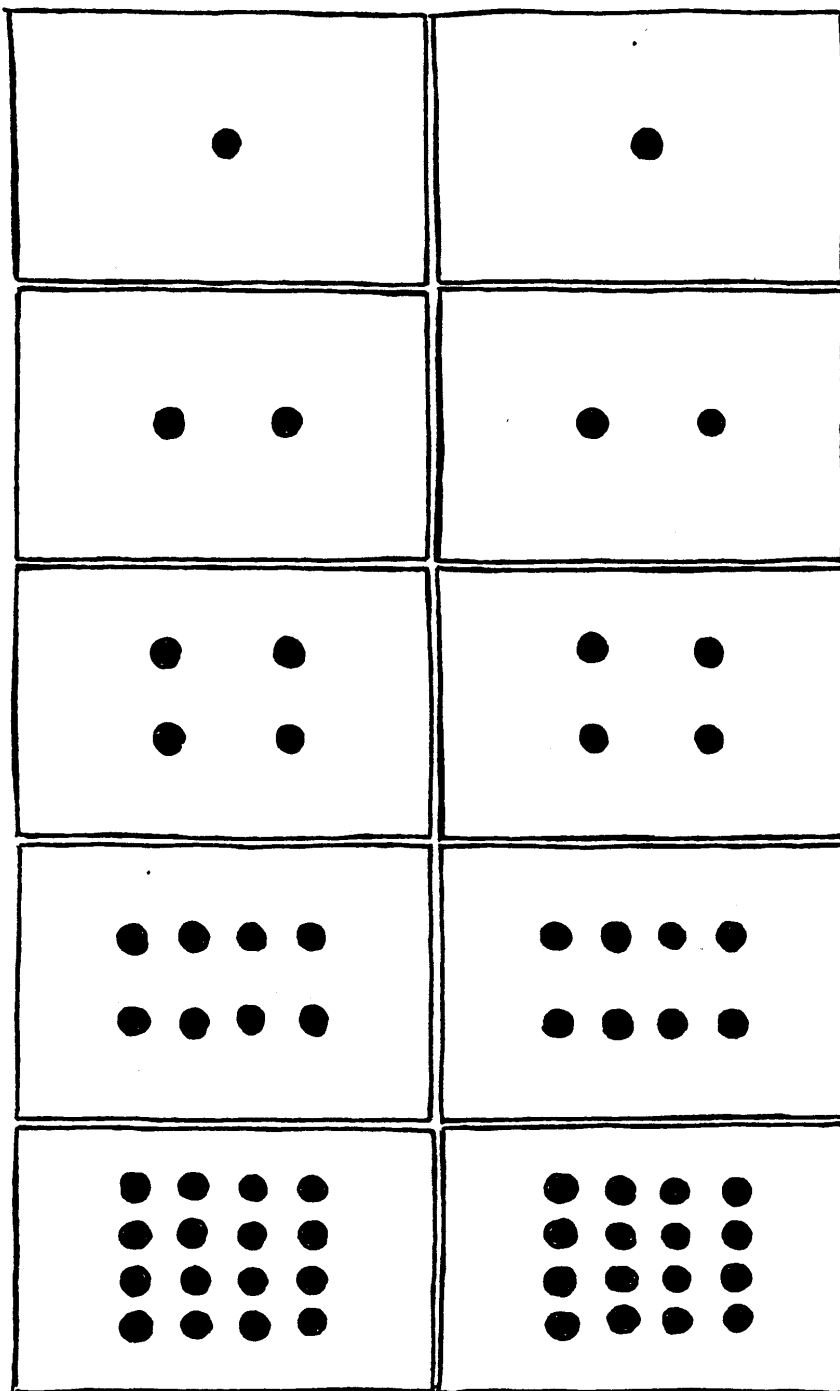
As well as representing numbers, this code can also represent the characters in a word-processor document. A byte is often used to represent a single character in a text—the numbers 0 to 255 are more than enough to encode all the upper- and lower-case letters, digits, punctuation, and many other symbols.

To represent larger numbers, several bytes are grouped together. Two bytes (16 bits) can represent 65,536 different values, and four bytes can represent over 4 billion values. The speed of a computer is affected by the number of bits it can process at once. For example, a 32-bit computer can perform arithmetic and manipulations on 32-bit numbers, whereas a 16-bit computer must break large numbers into 16-bit quantities, making it slower.

Normally we don't see the bits and bytes in a computer directly because they are automatically converted to characters and numbers when they are displayed, but ultimately bits and bytes are all that a computer uses to store numbers, text, and all other information.

Further reading

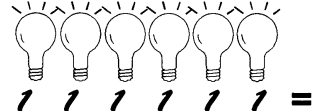
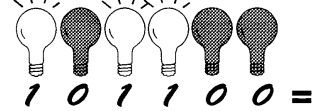
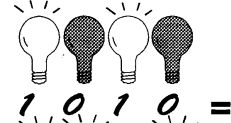
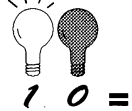
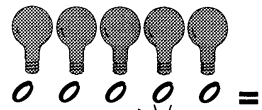
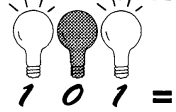
Most introductory computing texts discuss the binary number system. *My friend Arnold's book of Personal Computers* by Gareth Powell has a whole chapter on binary numbers.



Instructions: Copy this page onto card, and cut out the boxes to make two sets of five cards.

BINARY NUMBERS

Try to work out these binary numbers



a	b	c	d	e	f	g
1	2	3	4	5	6	7
h	i	j	k	l	m	n
8	9	10	11	12	13	14
o	p	q	r	s	t	u
15	16	17	18	19	20	21
v	w	x	y	z		
22	23	24	25	26		

Work out these secret codes



Instructions: Work out the numbers represented by the lightbulbs at the top of the page. Also, there is a message coded in binary at the bottom of the page; work out the numbers and look them up in the table to get the message.