CS Unplugged is a book of activities that illustrate computer science principles without using a computer.

Activities are short and are designed to be easily integrated into classes and include exercises and lesson plans for teachers.
• Data in computers is stored and transmitted as a series of zeros and ones.
  ▪ How can we represent words and numbers using just these two symbols?
What numerical property do you see in the dots on the cards?

Display the cards so the following number of dots are showing:

- 6
- 15
- 21
When a binary number card is **not** showing, it is represented by a zero. When it **is** showing, it is represented by a one. This is the binary number system (base 2).

- What are the following binary numbers?
  - $01001_2$
  - $10011_2$
What is the highest number we can represent using 5 cards?
- \(11111_2 = 31\)

What is the lowest number we can represent using 5 cards?
- \(00000_2 = 0\)

Count from 0 to 31 in binary.
• Letters are represented in computers in binary also!

<table>
<thead>
<tr>
<th>Letter</th>
<th>Value</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>blank</td>
<td>0</td>
<td>000000_2</td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>000001_2</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>000100_2</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>000110_2</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>26</td>
<td>11010_2</td>
</tr>
<tr>
<td>blank</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>N</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>O</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>P</td>
</tr>
<tr>
<td>D</td>
<td>4</td>
<td>Q</td>
</tr>
<tr>
<td>E</td>
<td>5</td>
<td>R</td>
</tr>
<tr>
<td>F</td>
<td>6</td>
<td>S</td>
</tr>
<tr>
<td>G</td>
<td>7</td>
<td>T</td>
</tr>
<tr>
<td>H</td>
<td>8</td>
<td>U</td>
</tr>
<tr>
<td>I</td>
<td>9</td>
<td>V</td>
</tr>
<tr>
<td>J</td>
<td>10</td>
<td>W</td>
</tr>
<tr>
<td>K</td>
<td>11</td>
<td>X</td>
</tr>
<tr>
<td>L</td>
<td>12</td>
<td>Y</td>
</tr>
<tr>
<td>M</td>
<td>13</td>
<td>Z</td>
</tr>
</tbody>
</table>
COUNT THE DOTS

HAPPY BIRTHDAY, CARLOS SANTANA!

Born July 20, 1947
• Computer screens are divided up into a grid of small dots called *pixels* (picture elements). In a black and white picture, each pixel is either black or white.

• Computers store drawings, photographs and other pictures using only numbers.

• The following activity demonstrates how a computer image can be stored efficiently.
• The letter a has been magnified to show the pixels. When a computer stores a picture, all that it needs to store is which dots are black and which are white.
COLOR BY NUMBERS

6,5,2,3
4,2,5,2,3,1
3,1,9,1,2,1
3,1,9,1,1,1
2,1,11,1
2,1,10,2
2,1,9,1,1,1
2,1,8,1,2,1
2,1,7,1,3,1
1,1,1,1,4,2,3,1
0,1,2,1,2,2,5,1
0,1,3,2,5,2
1,3,2,5
This technique is called run-length encoding.
  - Fax transmission
  - Compression of images

Color encoding
  - Use two numbers per run
    - First number is how many pixels as before
    - Second number is what color (1=red, 2=green, ...)

COLOR BY NUMBERS
When data is stored on a disk or transmitted from one computer to another, we usually assume that it doesn't get changed in the process. But sometimes things go wrong and the data is changed accidentally.

This activity uses a magic trick to show how to detect when data has been corrupted, and to correct it.
CARD FLIP MAGIC
This exercise illustrates even parity.

When computer data is transmitted to another computer, extra bits are added so that the number of 1s is even.

The receiving computer can detect if something gets messed up during the transmission and can correct it if there is one error.

What happens if there are two errors?
Here is an example of parity in real life:

1 x 10 = 10
4 x 9 = 36
2 x 8 = 16
5 x 7 = 35
9 x 6 = 54
3 x 5 = 15
7 x 4 = 28
6 x 3 = 18
7 x 2 = 14

\[ 226 \div 11 = 20 \text{ remainder } 6 \]
Checksum Digit = 11 - 6 = 5
• More parity:
• Since computers only have a limited amount of space to hold information, they need to represent information as efficiently as possible. This is called compression.
• By coding data before it is stored, and decoding it when it is retrieved, the computer can store more data, or send it faster through the Internet.
• This exercise illustrates how a children's rhyme can be compressed.
YOU CAN SAY THAT AGAIN!

PITTER PATTER
PITTER PATTER
LISTEN TO THE RAIN
PITTER PATTER
PITTER PATTER
ON THE WINDOW PANE
YOU CAN SAY THAT AGAIN!

PITTER PITTER PITTER PITTER

LISTEN TO THE RAIN

ON WINDOW Pane

BEFORE: 78 letters

AFTER: 29 letters
• The arrows and boxes are presented with 2 numbers.
• PITTER PA(7,4)
  ▪ 7: count back 7 positions
  ▪ 4: copy 4 letters/spaces
• Sometimes boxes point back to a box with a blank inside.
• The storage capacity of computers is growing at an unbelievable rate.
  - In the last 25 years, the amount of storage provided on a typical computer has grown about a million fold.
• We can *compress* the data so that it takes up less space.
  - This exercise uses Ziv-Lempel coding, or LZ coding, invented by two Israeli professors in the 1970s.
  - ZIP files, GIF images
• Computers are usually programmed using a "language", which is a limited vocabulary of instructions that can be obeyed.
• One of the most frustrating things about programming is that computers always obey the instructions to the letter, even if they produce a crazy result.
• This activity gives kids some experience with this aspect of programming.
MARCHING ORDERS
MARCHING ORDERS
• This activity illustrates structures used in parallel sorting networks.
• Kids sort data by walking through a sorting network laid out on the floor.
• The network simulates how a parallel network would sort data.
  ▪ Kids find out that data can be sorted a lot faster in parallel!
BEAT THE CLOCK
VIDEO

- csunplugged.org
- video.google.com
  - Search for “computer science unplugged”
TWENTY GUESSES

• How much information is there in a 1000-page book? Is there more information in a 1000-page telephone book, or in Tolkien's *Lord of the Rings*?
  ▪ If we can measure this, we can estimate how much space is needed to store the information.

• This activity introduces a way of measuring information content.
• Can you read the following sentence?

Ths sntnce hs th vwls mssng.

• You probably can, because there is not much "information" in the vowels.
• I am thinking of a number between 1 and 100.
• I will start you off with 20 pieces of candy.
• You may only ask questions that have a "yes" or "no" answer.
• For each incorrect guess, you will lose one piece of candy.
• Once you guess correctly, you can keep whatever candy remains.
TWENTY GUESSES

- To pick a number between 0 and 100, you only need 7 guesses.
  - Always shoot for the middle number of the range and eliminate half the possibilities!
  - This concept is called **binary search**.

- If the number was between 0 and 1000, you would only need 3 additional guesses.

- You can guess a number between 0 and 1 million in only 20 guesses!
• Computers are often required to find information in large collections of data.
• Computer scientists study quick and efficient ways of doing this.
• This activity demonstrates three different search methods so children can compare them.
BATTLESHPIS

- Battleships are lined up at sea.
- Each battleship has a number that is hidden.
- How many guesses does it take for you to find a specific battleship?
  - The number of guesses is the child's score.
  - The lowest score wins.
GAME 1: Ships are randomly ordered.

<table>
<thead>
<tr>
<th>Your Ships</th>
<th>Number of Shots Used:</th>
</tr>
</thead>
<tbody>
<tr>
<td>A B C D E F G H I J K L M</td>
<td></td>
</tr>
<tr>
<td>N O P Q R S T U V W X Y Z</td>
<td></td>
</tr>
</tbody>
</table>

FIND SHIP # 717
GAME 2: Ships are in increasing order.

<table>
<thead>
<tr>
<th>Your Ships</th>
<th>Number of Shots Used:</th>
</tr>
</thead>
<tbody>
<tr>
<td>33 183 730 911 1927 1943 2200 2215 3451 3519 4055 5548 5897 5905 6118 6296 6625</td>
<td></td>
</tr>
<tr>
<td>A B C D E F G H I J K L M</td>
<td></td>
</tr>
<tr>
<td>5785 5897 5905 6118 6296 6625 6771 6831 7151 7806 8077 9024 9328</td>
<td></td>
</tr>
<tr>
<td>N O P Q R S T U V W X Y Z</td>
<td></td>
</tr>
</tbody>
</table>

FIND SHIP # 5897

2A
GAME 3: Ships are ordered into 10 groups based on a mystery function.

<table>
<thead>
<tr>
<th>Your Ships</th>
<th>Number of Shots Used:</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
</tr>
<tr>
<td>K</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
</tr>
<tr>
<td>N</td>
<td></td>
</tr>
<tr>
<td>O</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td></td>
</tr>
<tr>
<td>U</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td></td>
</tr>
</tbody>
</table>

FIND SHIP # 8417
These three games illustrate
- linear search
- binary search
- hashing

What is the maximum number of guesses required for each of these search techniques
- for 26 battleships?
- for n battleships?
Many optimization problems involve situations where certain events cannot occur at the same time (e.g. scheduling meetings and classes).

Coloring regions of a map with different colors is effectively the same problem as we will show in this exercise.
POOR CARTOGRAPHER

• Given a map, color each region with a color so that no two adjacent regions use the same color.

• If two region touch only at one point, they are not considered adjacent.

  ▪ Example: Arizona and Colorado

• How many colors are sufficient to color a map?

• How many colors are necessary to color a map?
POOR CARTOGRAPHER
POOR CARTOGRAPHER
POOR CARTOGRAPHER
POOR CARTOGRAPHER

"undirected graph"
POOR CARTOGRAPHER

Scheduling
• Can we color a map with 3 colors?
• This problem is intractable.
• The only way we know to solve this problem in general is to derive all possible colorings and see if we come across a valid coloring.
• How many colorings are there for a map with 25 regions?
• How long would it take to analyze all of these colorings if it takes 1 second to analyze one coloring?
• When you have a lot of people using one resource (such as cars using roads, or messages getting through the Internet), there is the possibility of "deadlock".
  ▪ A way of working cooperatively is needed to avoid this happening.

• This exercise illustrates cooperative problem solving and (potentially) deadlock.
A shared resource in Pittsburgh:
THE COOPERATION GAME

- A shared resource in New York:
THE COOPERATION GAME

• Set up:
  ▪ Each child is assigned a color.
  ▪ Give two color cards to each child except one child, who gets only one.
    • Each child should not hold his or her own color initially.
  ▪ The children form a circle.

• Goal:
  ▪ Each child must end up with the card(s) of his or her own color.
THE COOPERATION GAME

• Passing Rules:

1. Only one card may be held in each hand.

2. A card can only be passed to an empty hand of an immediate neighbor in the circle. (A child can pass either of their two cards to their neighbor.)

3. (optional) No talking.
THE COOPERATION GAME

• Alternate Configurations

Diagram:

- Top row: 5 connected points
- Bottom row: 5 connected points
- Middle row: 3 connected points in a 3x3 grid
Routing and deadlock are problems in many networks, such as road systems, telephone and computer systems. Engineers spend a lot of time figuring out how to solve these problems - and how to design networks that make the problems easier to solve.
• Computer programs often need to process a sequence of symbols such as words in a document or even the text of another program.
• Computer scientists use a Finite State Automaton (FSA), a set of instructions to see if the sequence is acceptable or not.
• This exercise uses the FSA idea using treasure maps!
TREASURE HUNT

• Goal: Find Treasure Island, starting from Pirates' Island.
• Friendly pirate ships sail along fixed routes between islands offering rides to travelers.
• Each island has two departing ships, A and B.
• Determine all possible sequences of ships that a traveler can take to arrive at Treasure Island.
• Use your map to record all the ship routes.
TREASURE HUNT

Shipwreck Bay

A  B
TREASURE HUNT

Mutineers' Island

A  B
TREASURE HUNT
TREASURE HUNT

What is the quickest route?

"directed graph"
• Computers are often used to put lists into some sort of order (e.g. names into alphabetical order, appointments or e-mail by date, etc.)
  ▪ If you use the wrong method, it can take a long time to sort a large list into order, even on a fast computer.

• In this activity children will discover different methods for sorting, and see how a clever method can perform the task much more quickly than a simple one.
• Start with 8 containers with different amounts of sand or water inside. Seal tightly.
• Children are only allowed to use the scales to compare the relative weights of two containers.
• Only two containers can be compared at a time.
METHOD 1

• Weigh first two containers.
• Keep the lighter container and weigh it against the third container.
• Keep the lighter container and weigh it against the fourth container.
• ... continue ...
• Keep the lighter container and weigh it against the eighth container.
• Keep the lighter container. This must be the lightest container.
METHOD 1 (cont'd)

• Repeat this process again using the 7 remaining containers to get the next lightest container.
• Repeat this process again using the 6 remaining containers to get the next lightest container.
• Continue until all containers are sorted by weight.
• How may weight comparisons do we have to do to sort the containers?
• \[7 + 6 + 5 + 4 + 3 + 2 + 1 = 28\]
METHOD 2

• Take a random container and weigh it against every other container one at a time.
• Every container that is lighter is put into one group to the left.
• Every container that is heavier is put into another group to the right.
• Put the initially selected container between the two groups.
• Now sort the two groups in the same way.
METHOD 1 is called Selection Sort.
METHOD 2 is called Quick Sort.

Generally, quick sort is a lot faster than selection sort is.
For 8 containers, quick sort can reduce the number of weight comparisons to 13.
The basic edition of Computer Science Unplugged has 12 classroom exercises for you to use with your students.

Each exercise has a number of extensions, activities and background information.

All activities can be done without the use of computers, but they all demonstrate fundamental principles used in computers today.
The teacher's version of Computer Science Unplugged is available online at
http://www.csunplugged.org
- The book is FREE to download and use!
- Additional material will be published soon to add even more activities, including video to demonstrate how to use these activities effectively in your classroom.
10100 01000 00001 01110 01011

11001 01111 10101

(THANK YOU!)