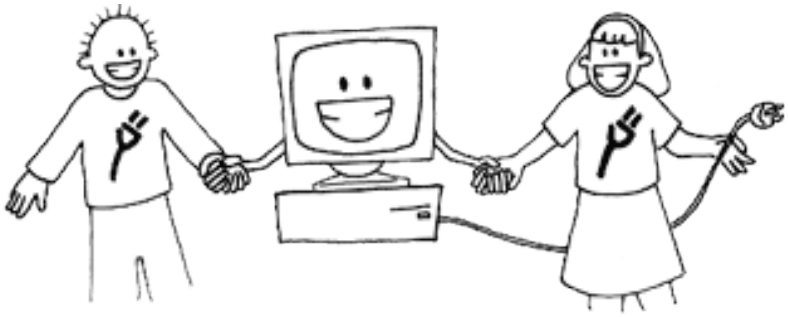
Computer Science Unplugged Activities

Compiled by Gail Carmichael (gail.banaszkiewicz@gmail.com)

CS Unplugged (http://csunplugged.org/) is "a collection of activities designed to teach the fundamentals of computer science without requiring a computer. Because they're independent of any particular hardware or software, Unplugged activities can be used anywhere, and the ideas they contain will never go out of date. Unplugged activities have been trialled and refined over 15 years in classrooms and out-of-school programmes around the world."



This document was written to provide an easy reference that may help determine which activity is best suited for your class. Keep in mind that computer science relates to many disciplines. After all, all fields have big problems to solve, and using computers is the best (and often only) way to solve big problems! So if you see something that might be related to the way problems might be solved in *your* class, don't feel limited by the suggestions made below. Most concepts can be extended and applied to just about any topic.

All activities will work for **grades seven to twelve**, and all activities will be well suited to **any level of computer science or technology course**.

A general explanation of what the field of computer science is all about would be a beneficial way to start things off. Any activity could be used for a whole class, possibly by including time for the volunteers to talk about their experiences at university and answer questions from students. Up to two activities could be done, especially for younger students who won't go into as much deep discussion.

In this document:

Table of Activities

Remember, all activities are suitable for grades seven to twelve, and they are all suitable for computer science and technology classes, do these won't be listed in the table.

Activity	Short Description	Additional Courses	Related Activities
1-Binary Numbers	How to represent numbers in base 2	Mathematics	
2-Image Representation	How raster images are digitally represented	Art	
3-Text Compression	Coding and decoding text passages for compression	Mathematics Biology	
4-Error Detection	A magic trick detects when data is corrupted and fixes it	Courses that deal with large amounts of data Mathematics	1-Binary Numbers
5-Information Theory	How to quantify the content of information	Grade 7/8 Science or Math	1-Binary Numbers
6-Searching Algorithms	How to store data so it can be found easily later	Mathematics	
7-Sorting Algorithms	How to efficiently sort data	Courses that deal with large amounts of data	
8-Sorting Networks	Sorting in parallel	Courses that deal with large amounts of data	7-Sorting Algorithms
9-Minimal Spanning Trees	Optimizing links between junctions in a network	Geography Younger Math	
10-Routing and Deadlock	Problems caused when competing for resources	Geography	8-Sorting Networks
11-Finite State Automata	Representing state and analyzing languages	Psychology English	
12-Programming Languages	Issues when giving precise instructions	Mathematics	
13-Graph Colouring	Colouring countries on a map with a small number of colours	Geography	
14-Dominating Sets	Marking dominating junctions in a network	Geography	13-Graph Colouring
15-Steiner Trees	Finding short paths through networks	Geography Younger Math	14-Dominating Sets
16-Information Hiding	Sharing information without actually revealing it	Younger Math History	17-Cryptographic Protocols 18-Public Key Encryption
17-Cryptographic Protocols	Agreeing on the outcome of a random	Younger Math History	16-Information Hiding 18- Public Key

	coin flip when you don't trust the other person		Encryption
18-Public Key Encryption	Locking a message with public info that only the intended recipient can unlock	Younger Math History	1-Binary Numbers 5-Information Theory 14-Dominating Sets 16-Information Hiding 17-Cryptographic Protocols
19-Human Interface Design	Human factors and issues in designing objects and software	Psychology	
20-The Turing Test	Determining if a computer is intelligent	Psychology	

Detailed Description of Activities

Activity 1: Binary Numbers

Additional Courses: Mathematics

This activity explores how to represent numbers in base two and the relationships and patterns found in numbers of base two. For older students, it could be extended to include analysis of other number systems, particularly hexadecimal. It is most suited to mathematics and computer science courses due to the strong analysis of number systems.

Activity 2: Image Representation

Additional Courses: Art

In this activity, students will learn how raster (pixel-based) images are stored in a computer. This could be extended to include an introduction to vector graphics (those that retain their quality when scaled) for more advanced students. Art students may be able to create their own examples or simply draw pictures in a style that looks like a pixelated computer image.

Activity 3: Text Compression

Additional Courses: Mathematics, Biology

Students will have an opportunity to play with various text compression algorithms by coding and decoding passages that are or will be compressed. Because this activity introduces algorithmic thinking, it is well suited to a math class. It may be possible to relate these concepts to compressing genetic data, thus relating the concept to a biology class.

Activity 4: Error Detection

Additional Courses: Any course that deals with a lot of data, Mathematics

Note: Students will get more out of this activity if they have done Activity 1: Binary Numbers first.

A 'magic trick' is used in this activity to show how one can detect when data has been corrupted, and how to fix it. It is well suited to any subject that deals with a large amount of data (such as experimental or social sciences), as it may help bring an appreciation of how delicate that data can be when stored on and transmitted with computers. Business classes might be particularly interested given how much they deal with numbers for accounting, etc.

Activity 5: Information Theory

Additional Courses: Grade7/8 Science or Math class

"Information theory provides a way of measuring information, and includes laws that define limits on how efficiently it can be stored or transmitted. This activity studies an important method of measuring information content." This activity could be combined naturally with Activity 1: Binary Numbers, since binary is a way of storing yes/no information.

Activity 6: Searching Algorithms

Additional Courses: Mathematics

Two popular ways to store data in a way that allows us to find it again quickly are demonstrated using a Battleship metaphor. The activity is suitable for math because of its algorithmic nature.

Activity 7: Sorting Algorithms

Additional Courses: Mathematics, Science

This activity uses actual balance scales to show how items might be sorted by weight. It compares a simple but slow algorithm with another still-simple but fast algorithm. It's good for math because of the algorithmic nature, and could be suitable for science classes, especially for the younger grades.

Activity 8: Sorting Networks

Additional Courses: Any course that deals with a lot of data

The focus of this activity is the use of parallel computers working together to sort large amounts of data faster than a single computer can. Activity 7: Sorting Algorithms is a useful activity to perform first, but not strictly necessary. The activity can be done on paper, but it is more fun to perform on a larger scale (marking the floor with tape, drawing on the ground with chalk...). There can be some good discussion on what types of problems are better solved by many people and what problems might not be helped by more.

Activity 9: Minimal Spanning Trees

Additional Courses: Geography, Younger Mathematics Classes

Minimum spanning trees are demonstrated as one way to optimize links in a network (of roads, telephones, etc) by studying an imaginary city that had no roads. The mayor wanted to decide where to pave the smallest number of roads possible. There are a lot of links to mapping, etc that could be discussed in a geography class.

Activity 10: Routing and Deadlock

Additional Courses: Geography

This activity shows how multiple people competing for resources can cause problems, and how messages can be routed to help ensure they get to their destination. The idea can be applied to designing efficient road or telephone networks. This activity could be performed with Activity 8: Sorting Networks, since deadlock is a common problem with parallel computing.

Activity 11: Finite State Automata

Addition courses: Psychology, English

A pirate's treasure map acts as a metaphor for finite state automata, which is a diagram that connects 'states' by allowed 'actions'. This concept can be viewed from two different perspectives: first, as a way

to capture the state of a situation and model behaviour (often used in artificial intelligence applications), and second, as a way of processing and analyzing languages.

Activity 12: Programming Languages

Additional Courses: Mathematics

Computers can only do what we tell them to do. This activity shows what problems can occur when trying to give someone exact instructions on how to do something. For more advanced math students, the functional programming language Scheme could be demonstrated, showing the connection between mathematical procedures, algorithms, and programming languages.

Activity 13: Graph Colouring

Additional Courses: Geography

This activity focuses on the problem related to colouring maps: countries sharing a border must be coloured differently, but we want to use as few colours as possible. It will also show how entities like maps can be represented as graphs (not the kind you plot in math class, though!).

Activity 14: Dominating Sets

Additional Courses: Geography

This activity discusses the problem of marking junctions in a network or graph in such a way that all other nodes are at most one step away from the marked ones. It turns out that this is actually a very difficult problem for the computer to solve. This activity follows naturally from Activity 13: Graph Colouring.

Activity 15: Steiner Trees

Additional Courses: Geography, Younger Math Classes

This activity shows how a small, seemingly insignificant variation on the problem described in Activity 9: Minimal Spanning Trees can make the problem rather harder to solve (as such, the activities would work well together). Steiner Trees are another way of trying to find short paths through networks, and the solution discussed here is actually algorithmically equivalent to those in activities 13 and 14.

Activity 16: Information Hiding

Additional Courses: Younger Math Classes, History

In relation to cryptography, this activity shows how information can be shared without actually revealing it. History students studying WWII would be interested in the role cryptography played in the war (e.g. the Enigma machine used by the Germans). The activity could be performed with the next two activities, also about cryptography.

Activity 17: Cryptographic Protocols

Additional Courses: Younger Math Classes, History

This activity shows how two people, who don't trust each other and are talking over the telephone, would be able to agree on the outcome of a random coin flip using a particular cryptography protocol. This activity could be performed with the previous and next activity, also about cryptography.

Activity 18: Public Key Encryption

Additional Courses: Younger Math Classes, History

Note: This is the most difficult activity to perform, and while it goes well with the previous two about cryptography, it may need an entire class to perform. It would probably help to have performed or seen examples from Activity 1: Binary Numbers, Activity 5: Information Theory, and Activity 14: Dominating Sets.

This activity shows how somebody can use public information to lock a message that can only be unlocked by the intended participant.

Activity 19: Human Interface Design

Additional Courses: Psychology

Students have the opportunity in this activity to think about the human factor when it comes to design, both for everyday objects and for computer software and hardware. The concept related heavily to psychology in that being able to think about human behaviour is very beneficial in finding a good design.

Activity 20: The Turing Test

Additional Courses: Psychology

The Turing Test was intended as a way to determine whether computers have the ability to exhibit intelligence. Students will see the test in action and be able to decide for themselves. Discussion can lead to the question of what intelligence is, and whether it's possible for computers to have it.