CS4FN

Computer Science for Fun

Issue 2

Claytronics: From goo to you

Exclusive Interview with Kevin Warwick. What is it like to be a Cyborg?

Enter the Future Human Competition

Special Issue on the

Magic of Computer Science

The magazine about the fun side of computer science

This issue is about magic, but not the Hogwart's kind, the computer science kind. The great thing about technology is that it is better than magic. Magic is about the impossible. Computer Science is about making the impossible routine. What about things that today are dismissed as fantasy…Is teleporting possible, or at least to be in two places at once? See "From goo to you" on Page 15. How about super-human powers? We can't do it unaided of course, but wire a human brain directly to a computer and some of it becomes a possibility. Find out what's already been done in Cyborg super-senses on page 6. We also look at how magician's tricks have led to new ways to build computers, and how your MP3 player is playing tricks on you. Microsoft kindly sponsored this issue, and since seeing into the future is always fun we start with a seeing-into-the-future quote from Bill Gates.

In this issue:

More in the cs4fn webzine www.dcs.qmul.ac.uk/cs4fn

- Magic of the sands – computer science transforms sand to sums
- Bugs, biology and computer science
- The mind-reading computer magician

Coming in the next issue:

- Robot wars - fighting with computer science
- Potter magic - how computer science made Hogwarts wizards work
- Big brother - computer science and telecommunications: I'm comming to get you

Future Proof

Bill Gates believes CDs and DVDs have had it. It won't be long before the whole back catalogue of music fits on a device in your pocket:

"It's going even faster than we expected…Five years from now people will say 'What's a CD? Why did you have to go to the case and open something up and you couldn't sequence it your own playlist way?' That will be a thing of the past. Even videos in the future will either be on a disk in your pocket or over the Internet, and far more convenient for you."

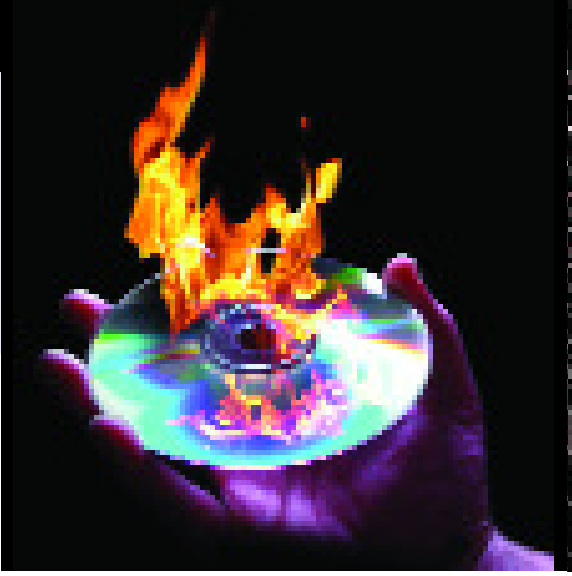
Bill Gates, Chairman and Chief Software Architect, Microsoft.

There are more predictions about future technology on the cs4fn website.

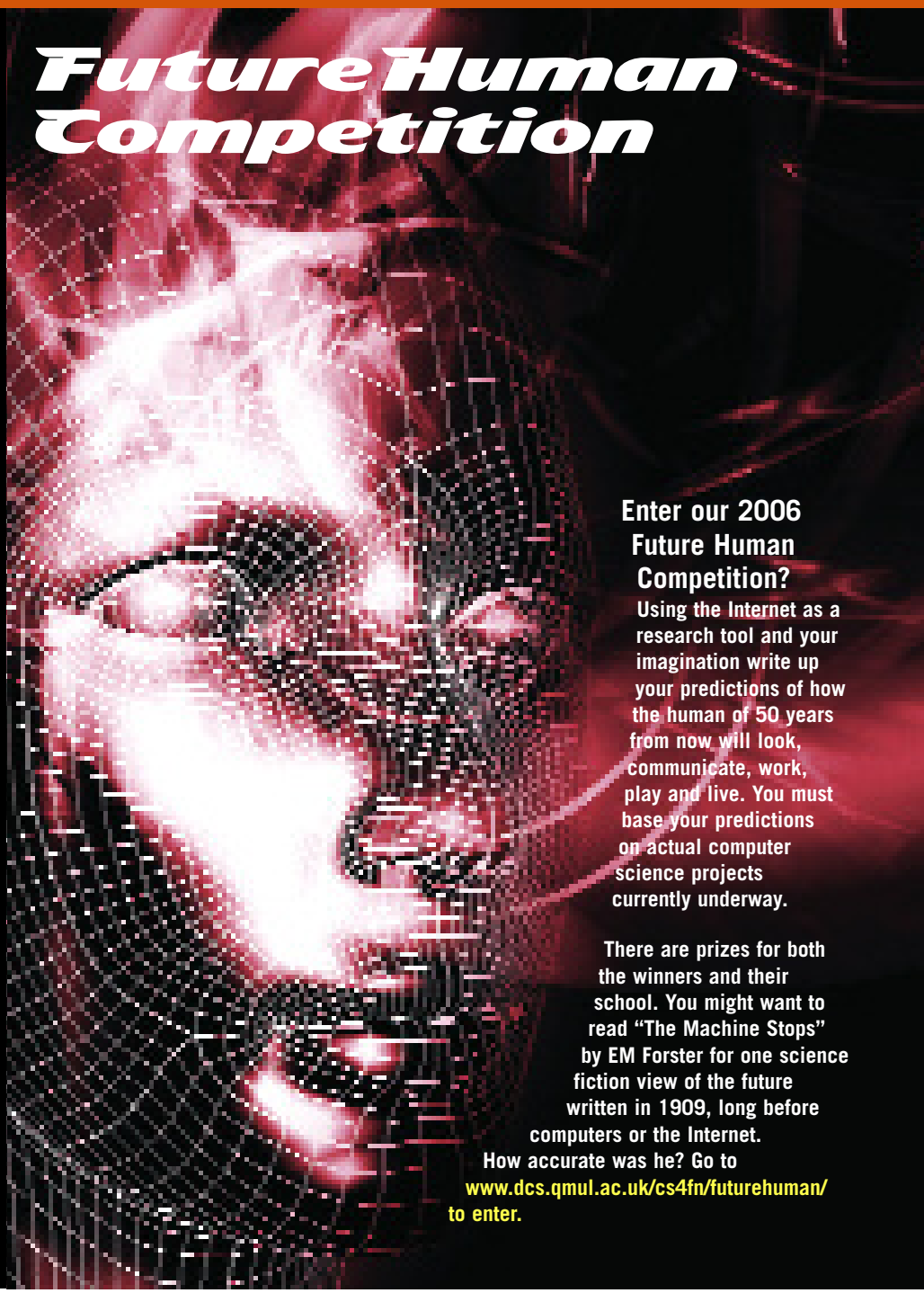
What do you think is most likely to disappear next?

Have your vote on the cs4fn site...

- Fixed phones?
- Cables?
- Written signatures?
- Loose change?
- Wrist-watches?
- Paper?
- Physical shops?
- Calculators?
- Radios?



Future Human Competition



Enter our 2006 Future Human Competition?

Using the Internet as a research tool and your imagination write up your predictions of how the human of 50 years from now will look, communicate, work, play and live. You must base your predictions on actual computer science projects currently underway.

There are prizes for both the winners and their school. You might want to read "The Machine Stops" by EM Forster for one science fiction view of the future written in 1909, long before computers or the Internet. How accurate was he? Go to www.dcs.qmul.ac.uk/cs4fn/futurehuman/ to enter.

BrainAcademy: The Compute-Ability Competition is Back

BrainAcademy the Compute-ability competition with the chance to win degree places as well as other career-enhancing plug-ins is back.

This year we will also be offering prizes for younger entrants who are not eligible to start University in 2007 – the chance to spend an afternoon in our Augmented-Human Interaction Lab. and find out first-hand some of the research going on in Computer Science. The theme this year is "Computer Science and the Entertainment Industry". For more details see the BrainAcademy website **www.brainacademy.qmul.ac.uk/**

This year there are prizes for younger entrants

Snakey Bites Back

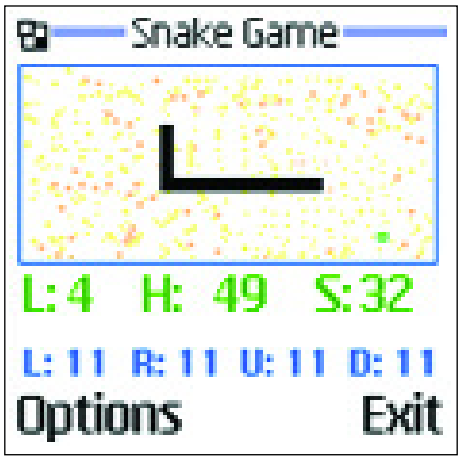
QGames is a fun new way of getting games for your mobile phone. It uses artificial intelligence (AI) techniques know as genetic algorithms to breed a virtual zoo full of evolved versions of the Snake game rated for how difficult a human player will find them. To rank the difficulty levels developer Milan Verma, a student at Queen Mary, University of London built a game playing computer program with 'human like' abilities. It 'plays' the game to decide its difficulty. It is based on data from real human game players to give it realistic properties.

The genetic algorithm uses a 'fitness function', an approximate way of working out what level of game playing talent the user has. It's a case of survival of the fittest - as with Darwin's natural selection that drives evolution in the natural world. Only the best fitting solution for each level of difficulty is selected.

A real user plays a few games so the AI can get a measure of their ability. Milan's program then sends them a game tailored to their game playing ability. It tailors Snake for each individual's needs by changing game play factors like speed, snake camouflage, environment and snake mobility. QGames is the first time this idea has been used in a mobile phone.

Milan has bred a zoo full of evolved versions of a game

To learn more about how QGames works and to get two free sample games, visit the Qwacky site linked from **www.dcs.qmul.ac.uk/cs4fn/** Updates are available via WAP and GPRS technology.



"QGames takes the human versus machine competition straight into your pocket; it's great fun to play and you never know what trick the AI will try to use to defeat you next time!"

Harry Potter's Invisibility Cloak

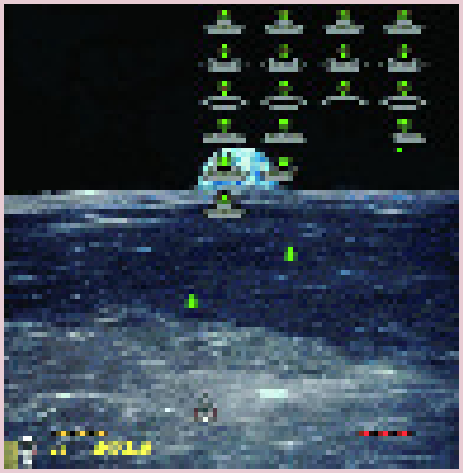
Harry Potter's invisibility cloak is surely Hogwarts' magic that science can't match. Wrap it round you and people just see through you as though you weren't there. Turns out even that kind of magic can be done with a combination of materials science and computer science. Professor Susumu Tachi of the University of Tokyo has developed a cloak made of thousands of tiny beads. Cameras video what is behind you and a computer system then projects the appropriate image onto the front of the cloak. The beads are made of a special material called retro-reflectrum. It is vital to give the image a natural feel – normal screens give too flat a look, losing the impression of seeing through the person. Now you see me, now you don't at the flick of a switch.

Pilot Error and Space Invaders

Another plane crashes with all on board killed. The papers blame the pilot. The official report agrees. It was pilot error again. But was the pilot really to blame?...and what does this have to do with cash machines? Would the pilot have still made the mistake if the cockpit - a very complicated computer interface - had been designed differently? Or is better training the solution?

This is something that is important for computer scientists to know - as if it is the design of the human-computer interface that is the problem then they can do something about it. You can help us find out, just by playing a Space Invaders game. Interested in helping? Find out more on the cs4fn website.

Join the experiment. Play the game.



The Texting Marrakech Game

How to Play

The aim of the texting Marrakech game is to have texted 3 numbers that add up to 15 before the other player. Each player takes it in turn to text the other a number from 1 to 9. Text a number already texted and you immediately lose.

Here's an example game

You text:	**8**
Your opponent texts:	**5**
You text:	**2**
Your opponent texts:	**4**
You text:	**6**
Your opponent texts:	**7**
You text:	**1**

and win
with **8, 6** and **1** = **15**
Notice that if you hadn't texted **6**, the opponent could have done and won with **5,4** and **6** = **15.**

The Secret

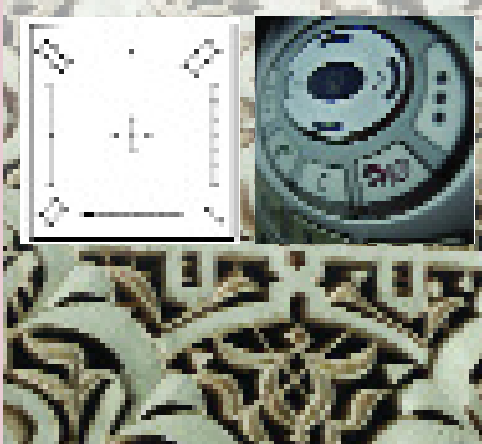
The Marrakech Texting game seems to need you to be amazingly clever at mental maths. In fact all you need is to know a simple bit of mathemagic: a magic square. A magic square is just a square of numbers where each row, column and diagonal add up to the same number.

You can use an ancient magic square known as the Loh Shu to help you win the Marrakech Texting Game. Find out how in the cs4fn webzine:
www.dcs.qmul.ac.uk/cs4fn/

Did you know?

The main town square in Marrakech is called the Magic Square

Magic Squares were invented in China. The earliest one so far discovered is on a scroll from 2800 BC.



Conjuring Cyborg Super Senses:

Exclusive Interview with Cyborg Kevin Warwick

Could humans ever gain super-human powers? Could the blind see with the help of computers? This is the stuff of science fiction comics and films - like Dr Octopus and the X-men or Geordi La Forge from Star Trek who could see using implants in his temples. Can computer science make things like telepathy, dismissed as magic today, become a reality of the future? We talked to the world's first Cyborg, Kevin Warwick about what it is like to have super-human senses.

Our senses are not great compared to other animals. Peregrine falcons can see small prey up to 5 miles away. Polar bears can smell a seal from similar distances. Many animals have senses we do not possess at all, like bats that can sense in ultrasound.

Inventing tools to improve our senses is almost a defining feature of being human. After all that is all that a pair of glasses does. Computer technology can even let us "see" things other than light. Radio telescopes, for example, let us see distant galaxies and the combination of the Internet and spy satellite technology allows everyone to look down from space on any street in the world.

All of this technology is disconnected though. It doesn't really give us superhuman senses any more than sitting in a tank gives us armour-plated skin. The technology turns the signals into something our limited abilities can sense. We then just use our eyes or ears as normal.

Neuroscientists increasingly understand how our brains and nervous system, the brain's communication system to the rest of the body, works. Signals from our eyes or fingertips pass down our nerves to the brain that turns them into understanding of the world around us. Similarly the brain sends messages out down the nerves to control our movements. Our brains are very adaptable, though. Where the messages come from originally or ultimately go to doesn't matter too much to the brain...and as they are basically just electrical signals computer technology can both detect them and recreate them.

"Sitting in a tank does not give you armour plated skin"

This leads to intriguing questions. What would be possible if we linked computers directly to the nervous system or even the human brain? Can we "cure" disabilities? Could we have super-human senses? Could we sense in ultrasound like a bat and what would it actually feel like? If the computers were linked to the Internet, could it literally take our senses further: outside our bodies altogether? Is any of this even remotely possible?

This all sounds like science fiction or magic but it has already happened. They are the kinds of question Cybernetics Professor, Kevin Warwick of Reading University (www.kevinwarwick.org/) is interested in...and he actually turned himself into the world's first cyborg to find out answers.

A team of neuroscientists implanted a chip directly into the nerves of his arm. It sensed the electrochemical signals travelling along his arm from his brain and transmitted them to a computer. The computer could also send signals to the implant. They then travelled along the nerves of his arm to his brain just like the normal sense signals when he touched something.

By connecting the computer to an ultrasound detector, with the signals from it being sent via the implant to his brain, he could literally feel objects moving towards him even when blindfolded. His brain learnt to interpret the signals from the ultrasound. He really had gained a completely new sense.

The implant was also connected to the Internet. For example, a robot arm copied his arm movements by sensing the signals between his brain and his hand as he flexed his fingers...even though he was in New York and the robotic hand in England.

What does it feel like? According to Kevin:

"Controlling a robot hand on a different continent with my own brain signals was really weird, especially when I could feel how much force the hand was applying. Even after thinking about this for some time I cannot fully imagine the extent of possibilities. Essentially your body can be any physical shape or size whatever that means."

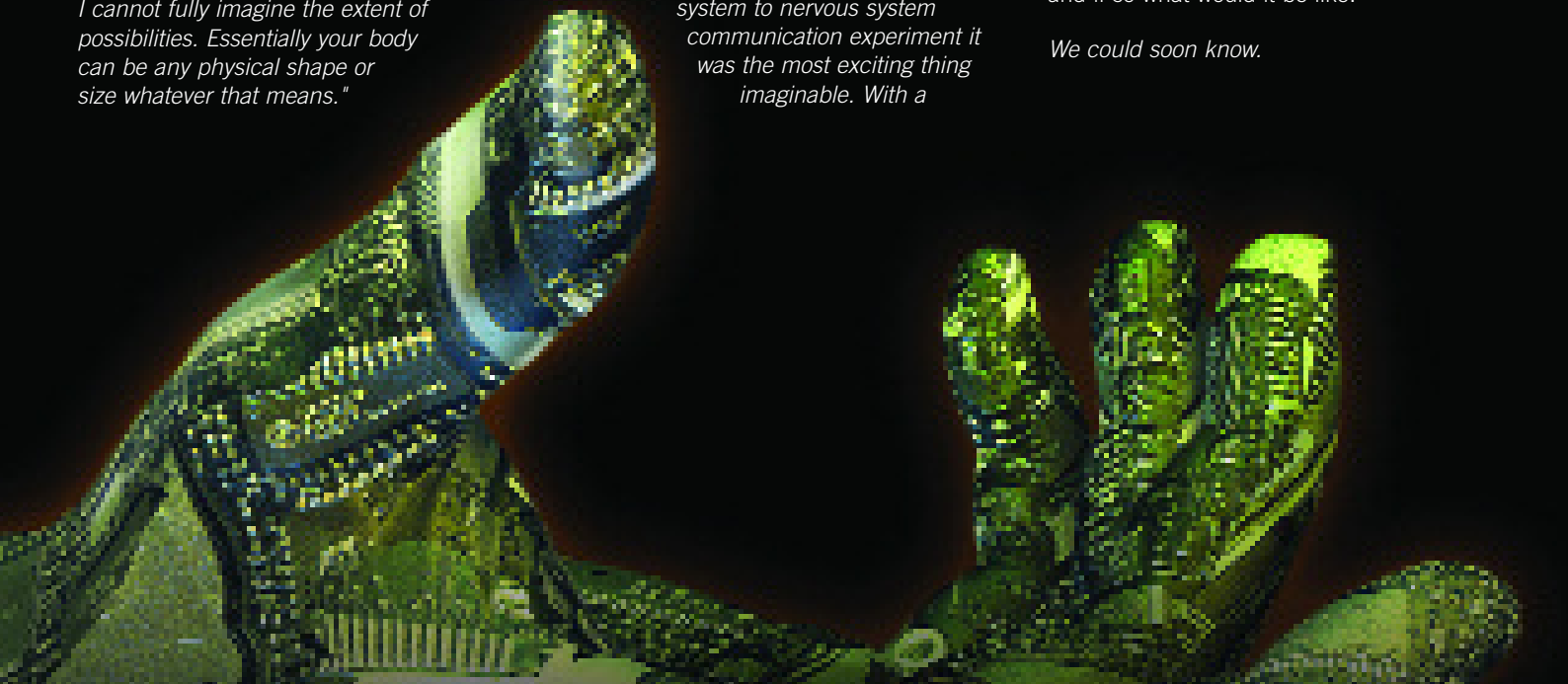
A student made a light-necklace for Kevin's wife, Irena, to wear. When he was relaxed the necklace sensed this via the implant over the Internet and was a cool blue. When he was excited however, his wife would know as her necklace glowed red as a result...even when she was in London and he was still in Reading. When she later had probes inserted into her arm, she didn't need to watch the necklace. She could directly feel the signals from his arm and vice versa, allowing them to communicate directly nervous system to nervous system.

Kevin is excited as to where this could lead:

"One of my heroes is Alexander Graham Bell, so when my wife and I succeeded with the direct nervous system to nervous system communication experiment it was the most exciting thing imaginable. With a blindfold on, my brain received neural pulses that originated from my wife's nervous system. Who knows where this will lead? Hopefully to thought communication, so maybe we won't need speech in the future."

The next step is to place implants not in the arm but directly in the brain. The use of brain implants is actually already quite widespread to help some people with Parkinson's disease live a normal life again. What hasn't been done is to connect brain implants to the Internet and so connect human's brains directly together. That raises even more amazing questions that were previously only the realm of science fiction and pseudoscience. Can more advanced forms of this technology make "telepathy" a reality? Could a human directly sense the thoughts of another, and if so what would it be like?

We could soon know.



Its good to talk:

Alexander Graham Bell

The famous inventor of the telephone Alexander Graham Bell was born in 1847 in Edinburgh, Scotland. His story is a fascinating one, showing that like all great inventions, a combination of talent, timing, drive and a few fortunate mistakes are what's needed to develop a technology that can change the world.

A talented Scot

As a child the young Alexander Graham Bell, Aleck, as he was known to his family, showed remarkable talents. He had the ability to look at the world in a different way, and come up with creative solutions to problems. Aged 14, Bell designed a device to remove the husks from wheat by combining a nailbrush and paddle into a rotary-brushing wheel.

Family talk

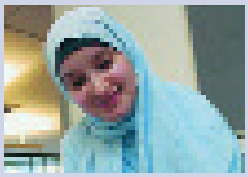
The Bell family had a talent with voices. His grandfather had made a name for himself as a notable, but often unemployed, actor. Aleck's mother was deaf, but rather than use her ear trumpet to talk to her like everyone else did, the young Alexander came up with the cunning idea that speaking to her in low, booming tones very close to her forehead would allow her to hear his voice through the vibrations his voice would make. This special bond with his mother gave him a lifelong interest in the education of deaf people, which combined with his inventive genius and some odd twists of fate were to change the world.

A visit to London, and a talking dog

While visiting London with his father, Aleck was fascinated by a demonstration of Sir Charles Wheatstone's "speaking machine", a mechanical contraption that made human like noises. On returning to Edinburgh their father challenged Aleck and his older brother to come up with a machine of their own. After some hard work and scrounging bits from around the place they built a machine with a mouth, throat, nose, movable tongue, and bellow for lungs, and it worked. It made human-like sounds. Delighted by his success Aleck went a step further and massaged the mouth of his Skye terrier so that the dog growls were heard as human words. Pretty ruff on the poor dog.

Speaking of teaching

By the time he was 16, Bell was teaching music and elocution at a boy's boarding school. He was still fascinated by trying to help those with speech problems improve their quality of life, and was very successful in this, later publishing two well-respected books called *The Practical Elocutionist and Stammering* and *Other Impediments of Speech*. Alexander and his brother toured the country giving demonstrations of their techniques to improve peoples' speech. He also started his study at the University of



See the webzine for the story of how computer science student Lila Harrar was also inspired by a deaf friend... and she has ended up with a commercial product.

London, where a mistake in reading German was to change his life and lay the foundations for the telecommunications revolution.

A 'silly' German mistake that changed the world

At University, Bell became fascinated by the ideas of German physicist Hermann Von Helmholtz. Von Helmholtz had produced a book, *On The Sensations of Tone*, in which he said that vowel sounds, a, e, i, o and u, could be produced using electrical tuning forks and resonators. However Bell couldn't read German very well, and mistakenly believed that Von Helmholtz's had written that vowel sounds could be transmitted over a wire. This misunderstanding changed history. As Bell later stated, "It gave me confidence. If I had been able to read German, I might never have begun my experiments in electricity."

The time for more than dots and dashes

His dreams of transmitting voices over a wire were still spinning round in his creative head. It just needed some new ideas to spark him off again. Samuel Morse had just developed Morse Code and the electronic telegraph, which allowed single messages in the form of long and short electronic pulses, dots and dashes, to be transmitted rapidly along a wire over huge distances. Bell saw the similarities between the idea of being able to send multiple messages and the multiple notes in a musical chord, the "harmonic telegraph" could be a way to send voices.

Chance encounter

Again chance played its roll in telecommunications history. At the electrical machine shop of Charles Williams in the USA, Bell ran into young Thomas Watson, a skilled electrical machinist able to build the devices that Bell was devising. The two teamed up and started to work toward making Bell's dream a reality.
To make this reality work they needed to invent two things: something to measure a voice at one end, and another device to reproduce the voice at the other, what we would call today the microphone and the speaker.

The speaker accident

June 2, 1875 was a landmark day for team Bell and Watson. Working in their laboratory they were trying to free a reed, a small flat piece of metal, which they had wound too tightly to the pole of an electromagnet. In trying to free it Watson produced a 'twang'. Bell heard the twang and came running. It was a sound similar to the sounds in human speech; this was the solution to producing an electronic voice, a discovery that must have come as a relief for all the dogs in the Boston area.

The mercury microphone

Bell had also discovered that a wire vibrated by his voice while partially dipped in a conducting liquid, like mercury or battery acid, could be made to produce a changing electrical current. They had a device where the voice could be transformed into an electronic signal. Now all that was needed was to put the two inventions together.

The first emergency call

On March 10, 1876, Bell and Watson set out to test their new system. The story goes that Bell knocked over a container with battery acid, which they were using as the conducting liquid in the 'microphone'. Spilled acid tends to be nasty and Bell shouted out "Mr. Watson, come here. I want you!" Watson, working in the next room, heard Bell's cry for help through the wire. The first phone call had been made, and Watson quickly went through to answer it. The telephone was invented, and Bell was only 29 years old.

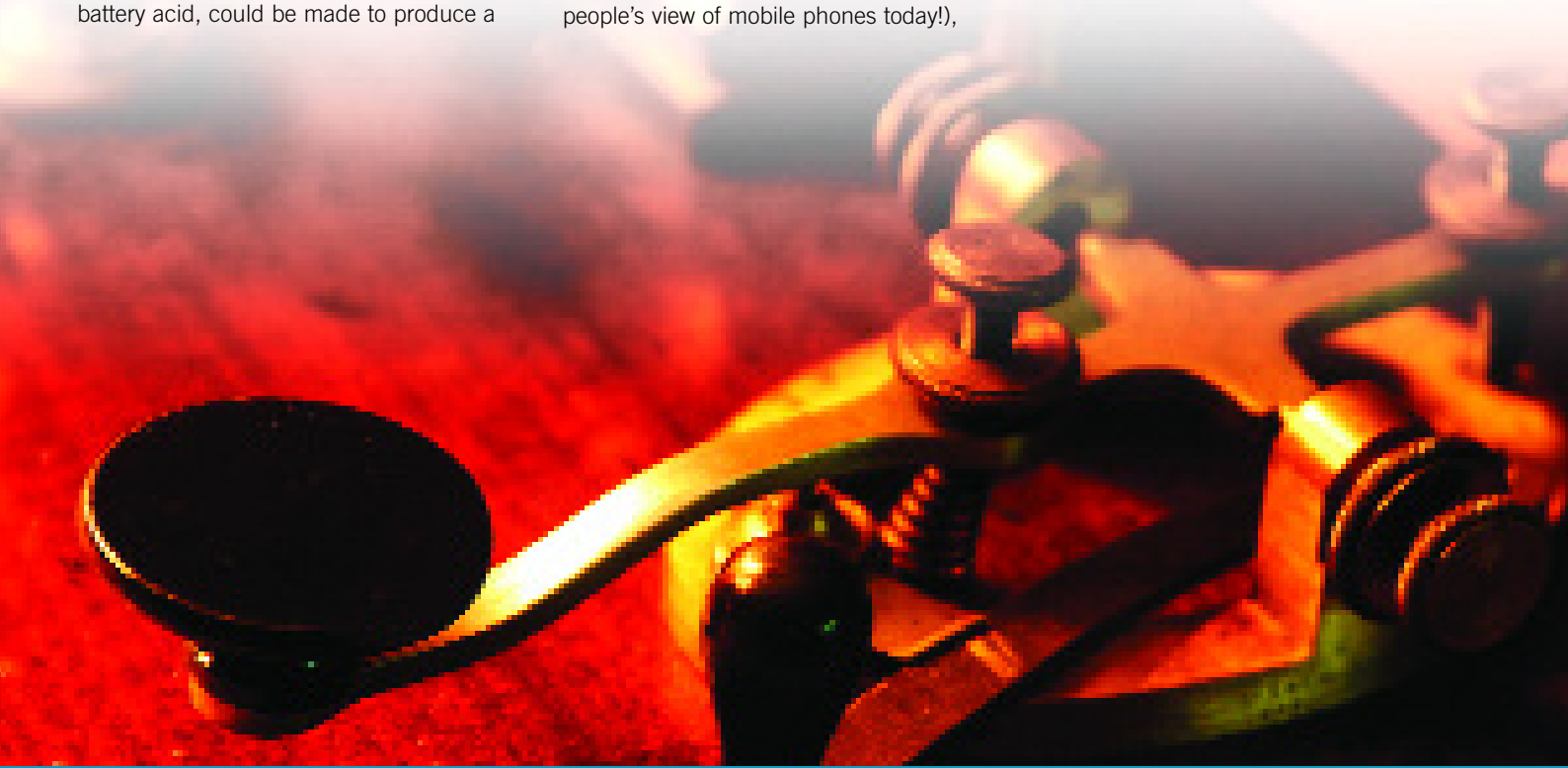
The world listens

The telephone was finally introduced to the world at the Centennial Exhibition in Philadelphia in 1876. Bell quoted Hamlet over the phone line from the main building 100 yards away, causing the surprised Brazilian Emperor Dom Pedro to exclaim, "My God, it talks", and talk it did. From there on, the rest, as they say, is history. The telephone spread throughout the world changing the way people lived their lives. Though it was not without its social problems. In many upper class homes it was considered to be vulgar. Many people considered it intrusive (just like some people's view of mobile phones today!), but eventually it became indispensable. Bell became rich and famous, and he was only in his mid thirties. The Bell telephone company was set up, and later went on to become AT&T one of Americas foremost telecommunications giants.

Can't keep a good idea down

Inventor Elisha Gray also independently designed his own version of the telephone. In fact both he and Bell rushed their designs to the US patent office within hours of each other, but Alexander Graham Bell patented his telephone first. With the massive amounts of money to be made Elisha Gray and Alexander Graham Bell entered into a famous legal battle over who had invented the telephone first, and Bell had to fight may legal battles over his lifetime as others claimed they had invented the technology first. In all the legal cases Bell won, partly many claimed because he was such a good communicator and had such a convincing speaking voice. As is often the way few people now remember the other inventors, though different countries now claim the invention of the telephone for different people, so there is plenty to talk about there!

Read Terry Pratchett's brilliant book* Going Postal *for a fun fantasy about inventing and making money from communication technology on DiscWorld.



Bad Wolf...

or a virus in your head?

Lights, cameras, web page

Back in the summer of 2001, the Steven Spielberg film AI: Artificial Intelligence was released. Along with the standard film trailers on the web came rumours about someone named Jeanine Salla. If you searched for the name you found a web page for a university scientist working on advanced robotics and artificial intelligence. It all looked very convincing: lists of scientific papers, a CV, and a full website for Bangalore World University. This was just the start. You could continue the trail and find other sites, drawing you into a world of robotic revolution. But the revolution wasn't about robots. The sites were fakes. The real revolution was the emergence of viral marketing as a tool for marketing movies.

The birth of Viral Marketing

Viral marketing is a different way to raise awareness of your film or TV show. You don't shove it forward using the usual posters or TV campaign. Instead you create a new online reality, and wait; someone will find it, and then they tell their friends, and their friends tell others. They post on bulletin boards and eventually you create an enormous buzz on the Internet. The low budget horror film *The Blair Witch Project* showed for the first time what a cost effective and powerful marketing strategy having a fictional web presence was. It was only a matter of time before this marketing method became mainstream. The fictional sites look as real as anything else on the web; there are layers of secrets and details, hidden text, web sites that look as if they have been 'hacked', with messages hidden in the source code of the web pages. These virtual worlds have the power to draw you in.

BBC's Bad Wolf

A recent example of good viral marketing is in the 2005 BBC TV series *Dr Who*, where clues on the mystery surrounding the identity of the 'Bad Wolf' were laid out across several fictional sites like www.badwolf.org.uk. There were fake sites that were actually referred to in the series (www.whoisdoctorwho.co.uk), a fake site for the UNIT group (www.unit.org.uk) and for the fictional company Geocomtex that appeared in the show (www.geocomtex.net) These sites were not publicised in the early stages. Their presence exploded onto the Internet as fans traded information. The viral marketing helped make the TV series a resounding success, and added a whole new dimension to the series.

A step too far?

In 2001 Electronic Arts released an innovative new game called Majestic. The conspiracy-based game invaded your life and couldn't be switched off; you give it your phone number, fax number and e-mail address. Strange faxes arrive, odd phone calls occurred at strange times and mysterious emails arrived. The line between game and reality blurred, as the designers made use of all the electronic forms of communication available to make the game play real. Surprisingly however the game was not a success. In fact it was shut down half way through costing the company many millions of pounds. In 2001, the game players of the world weren't ready for that level of immersive reality just yet.

Beyond the web

But there are still games around that are played by mixing fantasy with reality. For example Uncle Roy (www.uncleroyallaroundyou.co.uk/) is a game that links online players worldwide with players on the real streets of the city, who, using mobile computers search for the mysterious Uncle Roy. This type of entertainment is set to continue and expand as new technologies become available to give us new ways to communicate and play.

Will games of the future move off the screen and into people's lives, creating unique digital performances?

The future

The creative use of computers and the web in marketing films and television series through the creation of alternative realities will continue, and who knows where it will end. Perhaps in the not too distant future the immersive reality that Majestic pioneered may join up with the alternative realities created for viral marketing; a get together which could prove quite a show.

The next season of *Dr Who* will hit the TV screens as we go to press in Spring 2006. How will it be publicised? The Cybermen return: half human, half machine. Are they our future humans? We don't know how the publicity will go but suggest you look out for web links to Torchwood and Cybus Corporation. And K9 the Doctor's earthbound faithful robot dog, left with Sarah Jane Smith makes a surprise return: robot pets our future or our present? Let us know your views by entering our future human competition. See page 3.

Watch Out for the next cs4fn magazine: a special issue on Computer Science and Entertainment

Magical Memories just shuffling along

"Pick a card, any card!" How often have you heard magicians say that? The normal routine is that you pick a card, the magician shuffles the deck, and abracadabra, reveals your chosen card. But behind this magic often lies some interesting maths, and as we will see later, magicians' shuffles have actually led to the development of new ways for computers to work. Let's start with a trick to amaze your friends.

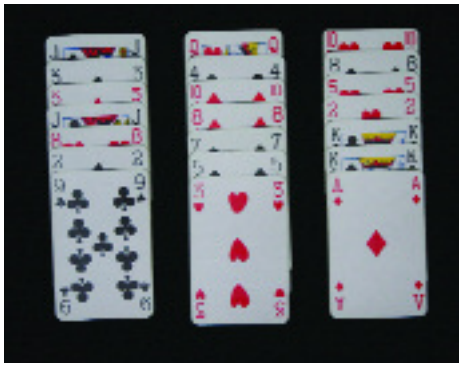
The 21-card trick.

The Magic effect: Have your friend shuffle a pack of cards and then you deal out single cards left to right into 3 piles of 7 cards, all face up. Your friend has to mentally select one of the cards. They mustn't tell you which card it is, but should tell you the pile it is in. You collect up the cards, and deal them out a card at a time left to right into three piles once more. Again they tell you the pile their card is in, you collect the cards once more, saying you're struggling to "read your friend's mind". Deal the cards out across the table in the three piles again in the same way. Your friend indicates the pile their card is in. Collect the cards again and deal them into the three piles one last time. You immediately announce their card and magically it is in the very middle position of the pack.

How do you perform the trick?

Let's look at the 'mechanics' of the trick. It involves several deals, each apparently shuffling the order of the cards, but doing so in a rather cunning way.

In fact it's really rather simple. All you have to do is make sure you always put the pile your friend selects carefully **between** the other two piles and deal the pack as above. Do that and after the fourth deal the middle card of the middle pile is the chosen card, which you can reveal as you see fit. Try and work out why it works, but then go to the cs4fn web site for an explanation.



Magic and Computers- developing your own algorithms

Once you understand the mechanics you can play with some ideas. The order of the chosen pile must not be changed, but the two other piles could for example be shuffled before being put together. As long as the chosen pile goes undisturbed between the two other piles of seven cards the order of the other cards doesn't matter. You might want to try and come up with your own additional twists now you know how it's done. The workings of this trick are what's known as an algorithm to computer scientists. The set of steps that you go through to get the trick to work are similar to the way that a computer steps through its instructions in a software program.

Brent Morris: Magician and Computer Scientist

The magicians' art of shuffling in special ways to make tricks, like the 21 card trick, work can also help us build computers. Magicians want to move cards around efficiently; computers want to move data around in their memory efficiently.

Perfecting the perfect shuffle

In a perfect shuffle, the magician cuts the cards exactly in half and perfectly interlaces them, alternating one card from each half. It takes years of practice to do but does look impressive. There are 2 kinds of perfect shuffles: With an out-shuffle the top card of the deck stays on top. With an in-shuffle the top card moves to the second position of the deck. Magicians know that 8 perfect out-shuffles restore the deck to its original order! It looks like the deck has been really mixed up, but it hasn't.

Computer scientist Brent Morris was fascinated by magic. In particular he became interested in the "perfect shuffle" in high school and has pursued its mathematics for more than 30 years with some amazing results. He earned his Doctorate in Maths from Duke University, and a Master's in computer science from Johns Hopkins University in the United States. He is believed to have the only doctorate in the world in card shuffling. He also holds two US patents on computers designed with shuffles, and has written a book on the subject called *Magic Tricks,*

A perfect shuffle in action

Card Shuffling, and Dynamic Computer Memories. Why the interest in perfect shuffles?

Binary shifts - as if by magic

You can use perfect shuffles to move the top card to any position in the pack, using a little bit of the maths behind computers: binary. Suppose you want the top card (let's call that position 0) to go to position 6. Write 6 in base 2 (binary), giving 110 (1x4+1x2+0x1). Now read the 0's and 1's from left to right: 1:1:0. Then, working through the 1's and 0's, you perform an out-shuffle for a 0 and an in-shuffle for a 1. In our case that means:

1: an in-shuffle, first

1: another in-shuffle,

0: and finally, an out-shuffle

As if by magic (if you are capable of doing perfect shuffles) the top card will have moved to position 6. Of course it works whatever the number, not just 6.

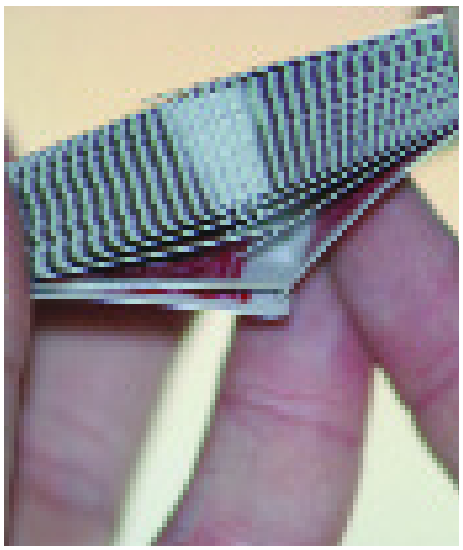
What does this have to do with the design of computers? You can use exactly the same ideas to move data efficiently around computer memory, which is what Brent Morris discovered and patented.

Curtain call

So as you impress your friends with your 21-card trick, coming up with your own performance ideas and are basking in that applause, remember two things.

Number one: a magician never reveals their secrets.

Number two: computers don't work by Hogwart's magic … only by mathematical magic.



Point, click and Sodarace

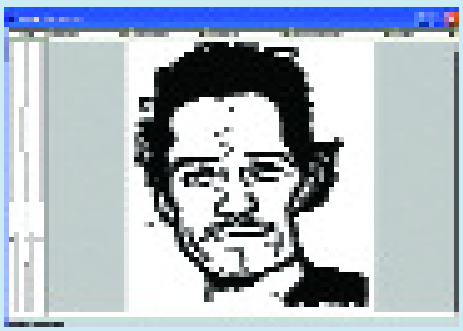
Following on from its successful 'run' at the Royal Society summer exhibition Sodarace, the online Olympics competition between human and machine creativity, has taken the next step forward.

Loads of people throughout the world have used Sodarace for their school and even university projects; from art and design to physics, chemistry, maths and biology, the options seem limitless and it's great fun.

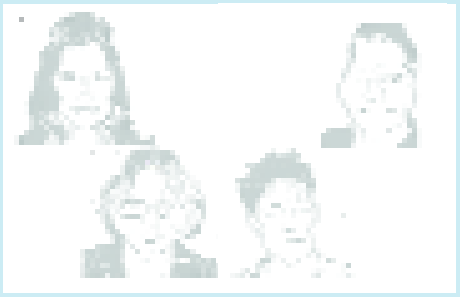
So to help make this even easier you can now download simple to use 'point and click' software free from **www.sodarace.net** that lets you select the racetrack terrain and the racers for your competitions. It's now straightforward to watch as your selected human and machine created racers go head to head; use your scientific skills to predict which will win on a particular racetrack (or just have a guess).

You can also download the first of a range of 'ready to use' lesson plans for classroom activities to try out. Why not ask your teachers to give them a go and let us know how you get on. Keep on racing.

Talk of the Toon



Orlando Bloom as a Soda Cartoon



It could be you!

Do you think computers can be creative like humans? They are good at repetitive tasks that are just too boring for humans to do, searching the web for pages on fashion tips or football scores. They can even play some games like chess better than humans. That's just done by searching through more possible moves than a person can. Does that count as creative? It doesn't seem, so. How about drawing? Cartoonists are creative people. If a computer could draw caricatures of people as well as human artists do, would that count as being creative? Queen Mary, University of London, computer science undergraduates, Lila Harrar and Akbar Hussain decided to find out. As part of their course, they created an Artificial Intelligence program based on the BAFTA award winning SodaConstructor, which draws cartoons of people. It works by choosing the most distinctive features of a face and exaggerating them just as human cartoonists do. At the Royal Society Summer Exhibition it was so successful they even had a Robot trying it out. You can get the AI to do a caricature of you at **www.dcs.qmul.ac.uk/cs4fn/** and find out how it sees you, then decide for yourself whether it is creative or not.

Sonic sleight of hand: The 'magic' of MP3

Believe it or not, your MP3 player is actually playing a sonic magic trick on you. MP3 is a format, a way of storing sounds and music so that they take up as little memory in the computer as possible. That way you can have hundreds of your favourite tracks on your iPod, which is really just a small computer disc that stores the digital music. The trick is to find a way to remove some of the sound information without your ears and brain noticing. The computer scientists who developed MP3 are playing a trick on your ears.

Sounds like music

So how is the trick done? Well first we need to look at sound and music. Sound is a change in air pressure. When Madonna sings, her vocal cords change the air pressure and the sound wave passes to our ears. Our eardrum converts this wave of air pressure into a mechanical movement in the tiny bones in our ear. That in turn is changed into a nerve signal that goes to our brain. A microphone works in the same way. The pressure wave moves a part of the microphone called the diaphragm, and it's this movement that turns the sound pressure wave into an electronic signal. When we are recording digital music we take many millions of very rapid samples - measurements - of the electrical signal and turn them into numbers to store.

Frequency asked questions

One of the things that characterises the sound wave is the frequency it is made of. Frequency means a regular repeating pattern. A very dull sound might just be a single frequency. The sound gets louder then softer, say every second, and repeats this cycle every second until you switch it off. This tedious sound would have a frequency of 1 Hertz (1 Hertz means one 'cycle' per second). Normal music doesn't sound at all like this dull repeating noise, but as it happens you can take any sound or music and, using some special mathematics, convert it into a set of different frequencies. Each of these frequencies alone just sound dull but added together it makes the music. The same idea is used in a music synthesiser; you can make the sound of any instrument by adding together the right frequencies. So we now know that our music can be described as frequencies, and we can start to play tricks.

Ear, ear

Turns out that our ears, though very clever, don't do everything well. Certain frequencies will stop you hearing other close by frequencies. These special frequencies 'mask' the presence of the others nearby. Once you know this (it was discovered through lots of experiments on hearing), then you know there is no point in using up valuable computer memory storing information on the frequencies your ears can't hear. So you don't store them. Kazam! You remove these frequencies altogether, but because they were masked by the other frequencies you can hear, you don't notice they have vanished.

Bunny in the headlights

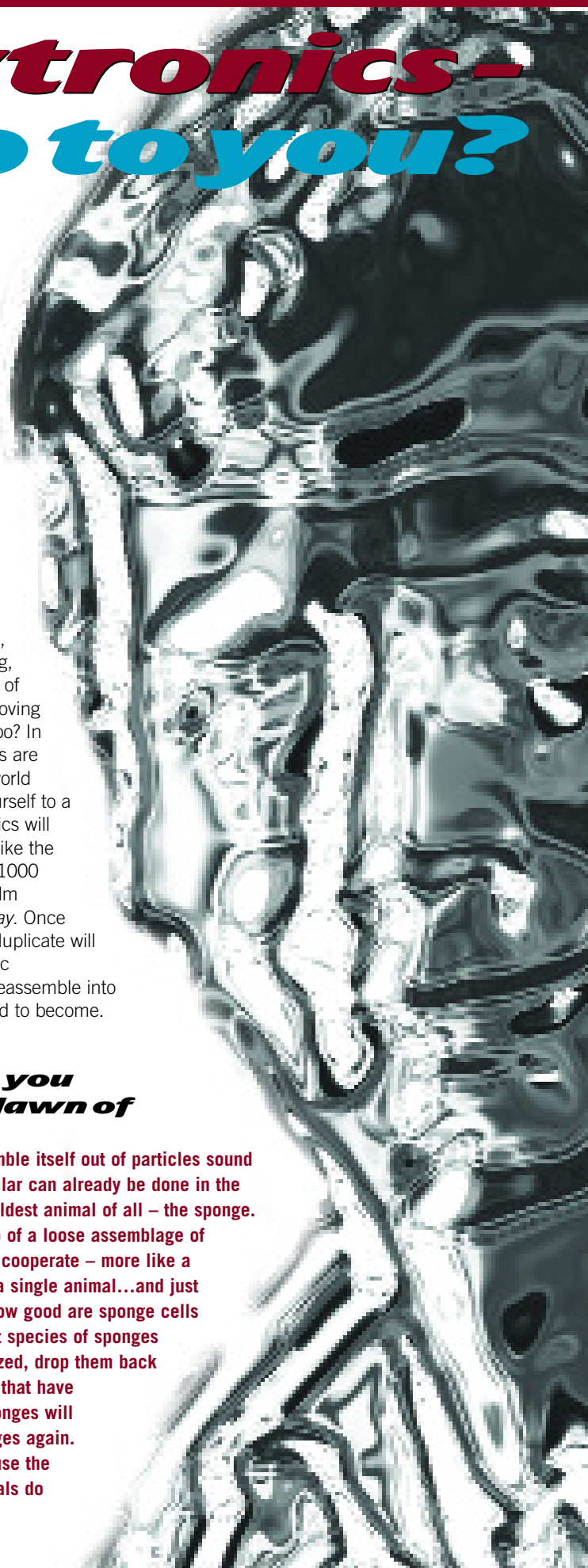
There is another cunning effect you can use. When we look at a bright light, for a while afterwards, we can't see a dim light. Our eyes change so we aren't dazzled by the bright flash, but this leaves our ability to see dimmer lights reduced forsoeome time. The same thing happens with sounds. A loud sound will stop you hearing a quieter sound that follows just after. So we can look at the digital music signal and work out all the places where a loud sound is followed by a quiet sound, and then cut the quiet sound out. This saves computer memory, and again the trick is that your ears won't notice the quieter sound is missing.

Sonic sleight of hand

It's using these tricks to vanish those parts of the sounds your ears won't notice are missing that gives MP3 its great ability to compress music. We need to store less data but the music sounds just as good. To be able to give the world MP3, Computer Scientists needed to work on ways to do the maths, as well as understand how our ears work. So the next time you listen to your MP3 player just think of all the maths and computing making up the wonderful din.

Yeh but no but... Read about compressing Vicky Pollard in the webzine

Claytronics – from goo to you?



It would be magical to be able to teleport: Star Trek's transporter was a clever device that allowed the characters to teleport from their spaceship to the planet. Devised by Gene Rodenberry, Star Trek's creator as a way to avoid the costly special effects of landing spaceships in the TV series, the transporter became the key to many an episode.

In effect, the transporter scans the passenger's atoms, disintegrating them into energy, then transmitting the energy to recreate a copy of that "pattern of atoms" on the planet below. It's a nice idea but the physics is frightening, disintegrating atoms and turning them into energy is exactly what a nuclear bomb does, and a nuclear bomb only converts a tiny fraction of matter to energy. If we really want to be able to create copies of ourselves at a distance we need to think of less explosive methods.

The physics is frightening ... exactly what a nuclear bomb does

It is just this problem that computer scientists in the USA at the Carnegie-Mellon University Synthetic Reality Project are looking at. Their solution: a new science called Claytronics. By using tiny programmable machines they hope to develop 'programmable matter', where millions of tiny devices called "claytronic atoms" or "catoms" would assemble into the shape of any object you want, connecting and disconnecting as they move. Currently they have developed experimental catoms that connect and move via magnets. These early devices are around four centimetres in size, much larger than the size the team want to develop, but you have to start somewhere.

The challenges are both in the technology, how do you build these tiny catoms, but also in the programming, how do you instruct billions of little machines to build a moving copy of yourself from the goo? In the future, if these problems are solved, we can imagine a world where you can transmit yourself to a meeting where the claytronics will build a copy of you, much like the smart liquid metal of the T-1000 Terminator android in the film *Terminator 2 Judgement Day*. Once your meeting is over, your duplicate will melt away and the claytronic programmable matter will reassemble into whatever it is next instructed to become.

From goo to you ... since the dawn of time

Does a robot that can assemble itself out of particles sound far-fetched? Something similar can already be done in the animal world...and by the oldest animal of all – the sponge. A sponge's body is made up of a loose assemblage of separate types of cells that cooperate – more like a colony of cells rather than a single animal...and just like the claytronics idea. How good are sponge cells at assembling? Put different species of sponges in a liquidizer. Once liquidized, drop them back into seawater and the cells that have survived of the different sponges will reassemble back into sponges again. Often computer scientists use the inspiration of the way animals do things to create advanced technology.

tlhingan hol dajatlh'a' (do you speak klingon)

Language is something we take for granted. We learn it as a child, maybe study a new language at school, or pick up some choice phrases for a holiday trip. A language consists of words - the bits it is made of - and grammar - the way these bits are put together. "The cat sat on the mat" makes sense. "Cat mat on sat the" doesn't. We have the same words but the rules of grammar are broken ... the information is lost. This confused list of words still follows a rule: the words are arranged alphabetically. But without the right rules, the rules of grammar, the words can't do what they need to do, which is let us know where that pesky cat is.

When we have to communicate with a computer and give it instructions we want to make life easy for ourselves. In the beginning programmers were forced to use binary – lists of 1's and 0's to tell the computer what to do. That's because computers are really just a very complex box of switches, and the 1's and 0's told the computer which parts of its circuits to switch on and off, but it made writing the programs a nightmare. It was asking you to speak in an alien tongue.

A better way to communicate with our computers

Something had to change, so those long-suffering programmers looked to what they knew best, human languages, to find a solution. In the same way as a word is made up from letters you could start to think of lists of binary instructions that you could associate together into a simple command to which you could attach some meaningful name. So rather than tell the computer in 1's and 0's ('0101010' say) to put a particular set of values into a particular circuit, you could use the phrase 'LET X=2' to mean the same. The computer understands how to turn a number into binary. It knows that 2 is 10 in binary (that means take a two and add no ones to it, just like a normal decimal 10 means take a ten and add no ones), so this useful idea of a computer language that WE understand means we don't need to talk the nasty alien computer lingo. The language instruction tells the computer to find a place in its memory, which the programmer wants to call X and to put 2 in there.

So to programming languages

The programmer's language is translated to the language the computer uses by following a list of instructions. In the same way as with a human language, if we get the grammar wrong and say typed '=LET 2 X' the computer wouldn't know what to do, the information is there but the grammar is wonky. So computer languages developed following many of the rules of human languages to try and keep it as easy as possible for human programmers to program. BASIC, Beginners All Symbolic Instruction Code was an early attempt. That's why, for example, there are university courses on Computer Science and Linguistics. The two are very closely related.

The universal translator?

There are now computer programs that translate human languages. You may have seen these on the Google or Babel Fish web sites for example. What is happening here is that, to translate say French text into German, the French is first translated to a language the computer understands and can work with. The text in that computer language is then translated back into German ... so it's passed through that strange alien binary language of the computer somewhere in between. Of course its very basic, Google just uses a simple approach to translation, but there are many researchers around the world trying to develop software that takes into account the complex rules of human languages and grammar. Some systems even try to learn languages from scratch like a child, in effect trying to produce the universal translator similar to the idea in Star Trek, which can translate any language to English. (Of course it would be easier to use the TARDIS telepathic circuits like Dr Who ... a small summer project there for anyone interested:-)

SYNTHETIC LANGUAGES

We tend to think of human languages as always having been around in their current form, and computer languages as new, but human languages need to have developed from somewhere, and that's another fascinating story where computers have helped. Some people have even created their own spoken languages such as Esperanto. The basic rules and words of Esperanto were proposed by LL Zamenhof at the end of the 19th century. The idea was to create a world language that everyone could speak. It never really took off, though there are still many people who learn it and use it.

Which brings us to Klingon, a made up language from Star Trek developed by Marc Okrand, (see the Klingon Language Institute at www.kli.org/). It has words, a fixed grammar and there are lots of 'native speakers' out there, so you guessed it ... there are programs that translate English to Klingon. It's interesting to think that a software program to translate a made up alien language does it by using its own made up alien language in the computer. As Mr Spock would say 'Quite Fascinating!'

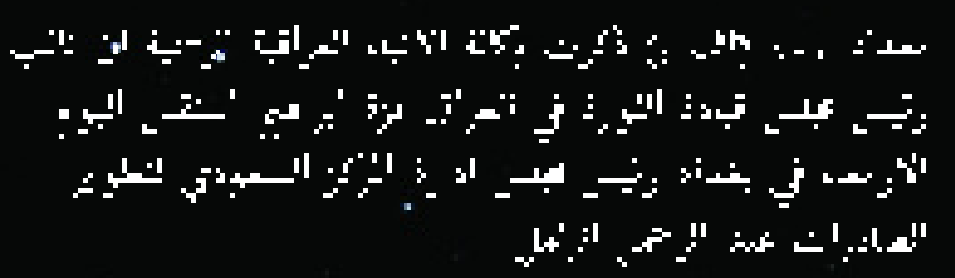
Qapla' ("Success!")

Footnote

In the interests of Galactic Peace it should also be pointed out that there are also programs that translate English into Vulcan, Romulan, Ewok, Wookie, Ferengi ...

Traditionally machine translation has involved professional human linguists manually writing lots of translation rules for the machines to follow. Recently there have been great advances in what is known as statistical machine translation where the machine learns the translations rules automatically.

It does this using a 'parallel corpus': just lots of pairs of sentences; one a sentence in the original language, the other its translation. Parallel corpora are extracted from multi-lingual news sources like the BBC web site where professional human translators have done the translations.



Let's look at an example translation:

Above is some original arabic with its translation below:

Machine Translation: Baghdad 1-1 (AFP) – The official Iraqi news agency reported that the Chinese vice-president of the Revolutionary Command Council in Iraq, Izzat Ibrahim, met today in Baghdad, chairman of the Saudi Export Development Center, Abdel Rahman al-Zamil.

Human Translation: Baghdad 1-1 (AFP) – Iraq's official news agency reported that the Deputy Chairman of the Iraqi Revolutionary Command Council, Izzet Ibrahim, today met with Abdul Rahman al-Zamil, Managing Director of the Saudi Center for Export Development.

This example shows a sentence from an Arabic newspaper then its translation by the Queen Mary, University of London's statistical machine translator, and finally a translation by a professional human translator. The statistical translation does allow a reader to get a rough understanding of the original Arabic sentence. There are several mistakes, though. Mistranslating the "Managing Director" of the export development center as its "chairman" is perhaps not too much of a problem. Mistranslating "Deputy Chairman" as the "Chinese vice-president" is very bad. That kind of mistranslation could easily lead to problems. That reminds me of the point in *The Hitch-Hiker's Guide to the Galaxy* where Arthur Dent's words "I seem to be having tremendous difficulty with my lifestyle," slip down a wormhole in space-time to be overheard by the Vl'hurg commander across a conference table. Unfortunately this was understood in the Vl'hurg tongue as the most dreadful insult imaginable, resulting in them waging terrible war for centuries....

For now the human's are still the best translators but the machines are learning from them fast!

Strictly X-Factor

TV talent shows like *X-Factor*, or *Soapstar Superstars* have always been popular. Its not just the talent on show that make them must see TV – it's having the right mix of personalities in the judges too. Simon Cowell has made a career of being rude – even reaching the dizzy heights of a guest appearance on *The Simpsons*. In contrast judge Sharon Osborne's on screen persona is far more supportive. It's often the tension between the judges that makes good TV.

However, if you believe Dr Who, the future of game shows will be robot judges like AnneDroid in the space age version of *The Weakest Link*…let's look at the robot future. How might you go about designing computer judges?

We need to write a program. We don't want to have to describe new judges from scratch each time. We want to do as little as possible to describe each new one.

What makes a judge

First let's describe a basic judge. We will create a plan, a bit like an architect's plan of a building. It can then be used to build individual judges. What's the X-factor that makes a judge a judge? First we need to decide on some characteristics of judges. We can make a list of them. The only thing common to all judges is they have different personalities and they make judgements on people. Let's simply say a judge's personality can be either supportive or rude, and their judgements are just marks out of 10 for whoever they are watching.

Character : SUPPORTIVE OR RUDE.
Judgement : 1 TO 10.

So let's start to specify (describe) Judges as people with a personality and capable of thinking of a mark.

DESCRIPTION OF a Judge:
    Character personality.
    Judgement mark.

All we are saying here is whenever we create a Judge it will have a personal character (it will be either RUDE or SUPPORTIVE). For any given judge we will refer to their character as "personality". It will also have a current judgement, which we will refer to as mark: a number between 1 and 10.

Best Behaviour

We are now able to say whether a judge is rude or supportive, but we haven't actually said what that means. We need to set out the behaviours associated with being rude and supportive. To keep it simple, let us say that the personality shows in the things they say. A rude judge will say "You're a disgrace" unless they are awarding a mark above 8/10. For high marks they will grudgingly say "You were ok I suppose".

TO Speak:
    IF (personality IS Rude) AND
        (mark <= 8)
    THEN SAY "You're a discrace".
    IF (personality IS Rude) AND
        (mark > 8)
    THEN SAY "You were ok I suppose".

It would be easy for us to give them lots more things to choose to say in a similar way. We can do the same for a supportive judge. They will say "You were stunning" if they award more than 5 out of 10 and otherwise say "You tried hard".

Ten out of Ten

The other thing that judges do is actually come up with their judgement. We will assume, to keep it simple here, that they just think of a random number – essentially throw a 10 sided dice under the desk with numbers 1-10 on. Judges' decisions can sometimes look like that on TV!

TO MakeJudgement:
    mark = RANDOM (1 TO 10).

Putting that all together to make our full judge description we get:

Our final plan for making judges

DESCRIPTION OF A Judge:
  Character personality.
  Judgement mark.

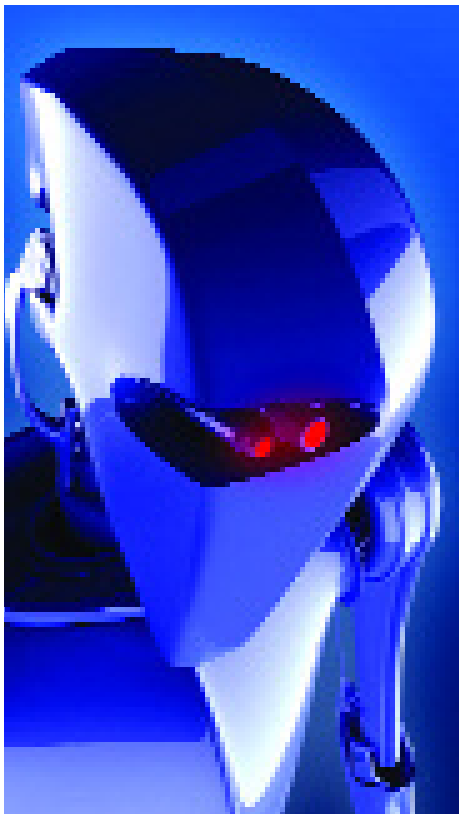
  TO Speak:
    IF (personality IS Rude) AND
      (mark <= 8)
    THEN SAY "You're a discrace".

    IF (personality IS Rude) AND
      (mark > 8)
    THEN SAY "You were ok I suppose".

    IF (personality IS Supportive) AND
      (mark > 5)
    THEN SAY "You were stunning".

    IF (personality IS Supportive) AND
      (mark <= 5)
    THEN SAY "You tried hard".

  TO MakeJudgement:
    mark = RANDOM (1 TO 10).



Kind words for our contestants?

Suppose now we want to create a rude judge, called SimonCoward. We can use the plan. We need to say what its personality is (Judges just think of a mark when they actually see an act so we don't have to give a mark now.)

SimonCoward IS A NEW Judge WITH
personality Rude.

This creates a new judge called SimonCoward and makes it Rude.
We could similarly create a rude AnneDroid:

AnneDroid IS A NEW Judge WITH
personality Rude.

For a supportive judge that we decide to call SharONN we would just say:

SharONN IS A NEW Judge WITH
personality Supportive.

Whereas in the specification we are describing a plan to use to create a Judge, here we are actually using that plan and making different Judges. So this way we can quickly and easily make new judge clones without copying out all the description again.

A classless society?

Computer Scientists are lazy beings – if they can find a way to do something that involves less work, they do it, allowing them to stay in bed longer. The idea we have been using to save work here is just that of describing classes of things and their properties and behaviour. Scientists do that a lot:

Birds have feathers (a property) and lay eggs (a behaviour).
Spiders have eight legs (a property) and make silk (a behaviour)

We can say something is a particular instance of a class of thing and that tells us a lot about it without having to spell it all out each time (even for fictional ones): eg

Hedwig is a bird. (so feathers and eggs)
Charlotte is a spider. (so legs and silk)

So we can now create judges to our hearts content, fixing their personalities and putting the words in their mouths based on our single description of what a Judge is.

All Change

We have specified what it means to be a robotic judge and we've only had to specify the basics of Judgeness once to do it. That means that if we decide to change anything in the basic judge (like giving them a better way to come up with a mark than randomly or having them choose things to say from a big database of supportive or rude comments) changing it in the plan will apply to all the judges of whatever kind.

What we have created is our first object-based program – it would be relatively easy to convert this into a program in a programming language like Java or C#.

We could create robot performers in a similar way (after all don't all the winners seem to merge into one in the end?). We would then also have to write some instructions about how to work out who won – does the audience have a vote? When do judges make judgements? When can they speak their mind? How many get knocked out each week? That's no harder. Why not give it a try and judge for yourself?

So how does a SharONN Judge have daughter KeLEE judges without any help from Ozzie? See **www.dcs.qmul.ac.uk/cs4fn/**

Back (page) to the drawing board.

You may have heard the term software engineering. Building a complex computer program is like building any complex machine or structure. It takes skill, professionalism, team-work and the ability to learn from your mistakes. Every type of engineering throughout history has had its share of disasters. Early steam engines exploded, the Titanic famously sank, the Hindenburg airship caught fire, and the Tay Bridge in Scotland fell down.

Software engineering is no different. Here are a few of its disasters to learn from (more in the cs4fn webzine):

The Ariane 5 Rocket (problems with big numbers in small spaces, 1996)

In 1996 an Ariane 5 rocket exploded forty seconds after lift-off. The project had taken 10 years, and cost $500 million. This spectacular software failure was due to squeezing a big number into the computer memory reserved for a small one. There wasn't enough space to hold the rocket's speed when it was passed to another smaller memory store. This caused the rocket to veer off course, break up and explode.

The moral: Make sure numbers fit their destinations.

Mars Climate Orbiter (a weighty problem in space, 1999)

Programmers work in teams to build software for a space mission. Unfortunately for NASA's $125 million Mars Climate Orbiter, two teams didn't know what the other was up to. One team was using Imperial weight measures (pounds). The other was using metric (kilograms). The mistake wasn't found until, when finally in space, the two programs spoke to each other, got very confused and caused the spacecraft to become lost in space.

The moral: software engineering is also about team communication

AT&T Crash (The day the phones stopped working...all of them, 1990)

In late 1989, AT&T engineers upgraded the software of their 114 US switching centres: the computers that make the connections so your phone links to the one you are calling. Each computer was duplicated so if one went wrong the other would take over. On January 15th 1990, they stopped working: 70 million calls failed. The problem was in a single line of code out of millions...and it was in both computers' copies. AT&T lost $1 billion as customers fled to their competitors.

The moral: with software, duplication doesn't always help

Big numbers can hurt

USS Yorktown Stops (a big something caused by nothing, 1998)

Dividing by zero is a bad idea. The answer doesn't exist. However a crewmember of the computer controlled guided-missile cruiser USS Yorktown mistakenly entered a zero on their console. It resulted in the computer program trying to do an impossible divide by zero. The program crashed and caused a failure in other linked computer systems on the ship, eventually shutting down the ship's engines, leaving it drifting for hours.

The moral: always check numbers are as expected.

The Pentium Chip Error (not enough numbers in the table, 1994)

The Pentium Chip used a look up table (LUT) to do division; basically it uses a pre-calculated set of numbers to speed things up. The LUT should have contained 1066 elements, but when the numbers were downloaded a bug in the software only put in 1061 of them. No one checked, and the chip went to manufacture with those numbers missing. When the mistake was found the chips all had to be replaced. The cost was more than $4 billion.

The moral: keep testing all the way.

Bugs can kill

The missing American Patriot Missile (a problem with bad timekeeping, 1991)

In 1991, during the first Gulf War, an American Patriot Missile battery in Dharan, Saudi Arabia, failed to shoot down an incoming Iraqi Scud missile. The Scud missile hit an army barracks with many casualties. State-of-the-art computers controlled the Patriot missile, but there was a problem. To work Patriot needed to accurately know the time. This was done with an internal clock that started to tick when the computer was first switched on. However the computer program, when converting the internal clock time into the time used by the guidance system introduced a tiny mistake. It rounded the number down slightly. With each passing second the error became larger until finally it was enough to make the missile miss.

The moral: small mistakes in calculations often build into big mistakes

And Finally expect the unexpected

In the early days of electronic computers they used relays, electromechanical switches that rocked up and down to switch the electrical circuits. Grace Murray Hopper, who was in charge of the team working with the Mark II computer, (an early electro-mechanical device), found that a moth had flown through the window and blocked one of the relays, so shutting the system down. This is arguably where the term computer 'bug' comes from.

The moral: The things some moths get into can be shocking!

---

cs4fn Paul Curzon, Peter McOwan, Gabriella Kazai and Christof Monz of the Department of Computer Science, Queen Mary, University of London. Keypad photos by Ferenc Kazai. Proof reading by Sue White.

EPSRC Engineering and Physical Sciences Research Council
Sodarace and the Human Error Modelling Project are funded by EPSRC

Microsoft
Microsoft provide support for printing.

Queen Mary University of London

Passionate about computer science?
www.dcs.qmul.ac.uk/cs4fn/