BITS AND BYTES

NEWSLETTER

6th

EDITION

INSIDE

BITCOIN
What is bitcoin?

STARTUPS
Uberification, Thousandeyes, Zenefits.

LEGENDARY PROGRAMMERS
Linus Torvalds

COVER STORY
AARON SWARTZ
Internet's own boy

TABLE OF
CONTENTS

EDITORIAL

TEAM

Head of Department
Dr. R. Leela Veluswamy

Faculty Advisor Comittee
Dr. R. Leela Veluswamy
Prof. R. Gururaj

Chief Editor
Akhila Yerukola

Co - Editor
Shruthi M
Prasann Dubey

Content Heads
Ashutosh Agarwal
Jayti Singh

Design Head
Anurag Kumar

Editors
Anugrahaa Ramesh
Keerthana Ganesan
R G Thivyavignesh

Content Team
Arvindmani
Ayush Mishra
Panigrahi
Parth
Rahul Kar
Sachin
Soorya N

Design Team
Lakshmanram
Manjith
Pradhiksha
Ritul
Shilpa

The Department of Computer Science & Engineering, NIT Tiruchirappalli, is ranked among the top 12 departments for Computer Science in India. It is also regarded as a safe haven for VLSI students. Being the most in-demand programme, a lot of expectations rest on the faculty, infrastructure and courses offered. As a result, the programmes offered have matured into some of the most highly recommended in the nation.

The pedagogical pattern for teaching computer programming in NIT Trichy is designed in such a way that students are at the core focus whilst teachers are provided a realistic chance to inculcate a vast amount of knowledge in the students. The department intends to strive for excellence in the coming years via a balanced process, where innovation and tradition grapple for dominance.

Bits & Bytes, the newsletter of the department, hopes to project the same modus operandi, by presenting the latest news, research avenues, programming hacks and heart-warming stories of people in love with computer science.

In this issue, we hope to entertain and enlighten you with pieces ranging from a feature on Linus Torvalds to a comparison report on startups, a simplified slice on hashing and the elephant in the room, Virtual Currency.

With hopes that our efforts reach out and leave a mark

Striving for excellence
0110001 01101000 01100101 01100101 01110010 0110011!

Sachin Ashok
Content

Basic Idea

Okay, so imagine you're at a table with some friends. You come up with a set of rules to create a currency that you can only use between each other.

Around the world today, there are lots of different types of currencies, and pretty much all of them have somebody in charge, be it a bank or a government or a company. But, being the sceptics that you are, you and your friends want to eliminate the guy in between the money and the consumers in favour of a more transparent process.

This is the idea behind Bitcoin, where instead of having one big guy in charge, millions of people all watch each other and make sure everything is fine.

Everyone knows how much money everyone else has because every time someone gives someone else some money, they both tell the whole world and it goes into the **Bitcoin Public Permanent Record**. More on that later.

Cool. You have an idea now, but you need a mechanism to implement this. So you guys chat a little bit more and decide that math is hard, but workable, given time and energy. You also decide that there's a certain math problem that all of you like to do and you set how many times you want to try to solve a really hard math problem as the basis for adding to the record i.e. when someone in your group solves that really hard math problem you'd decided on, that person tells the others of his success. The others can quickly verify that if the problem has indeed been solved, and after checking their work, they all write down that the person has gained some money. If the solution happened to be wrong, everyone just goes on with their life.

This is the central idea behind bitcoin mining.

Transactions

A transaction is when someone sends money to someone else. When you send money to someone else, you tell everyone else in the group to whom you're sending the money. However, this is done in a way that makes it really hard to actually tell who is sending and who is receiving. Like most things, it's not impossible to tell, but it would take someone a lot of time to figure it out.

CSE
APPLICATION

When someone sends money, everyone else checks their copy of the record to ensure that the person actually has enough money to send. This is especially important to the receiver. If enough people say that the sender doesn't have the money to send, the sender's transaction is ignored. He'll be able to try again in a little bit (a little sadder, a little wiser).

But what's to ensure that everyone else in the group actually cares enough to do the math and check?

Every entry in the Permanent Record records a bunch of people's trades, along with one extra trade of 50 coins that can go to anybody, made up of the transaction fees, a little extra amount the sender pays with every transaction. The rule was that whoever did the work that set up the entry to be good enough for the Permanent Record could give assign these 50 coins to anyone, including, you guessed right, himself!

A lot of people were excited by this idea and they started trading bitcoins for fun. Eventually, it got popular enough that some people started trading items and even real money for bitcoins.

Consequences?

Bitcoin is a network to transfer money/value in the same way Bit-torrent is a collaboration of users to transfer files. The idea is not to acquire Bitcoins and "cash out", but rather to build a truly global economy accessible to everyone.

Bitcoins are considered as currency because people agree that they are, the same way the Chinese think that Yuan is a currency or the Australians think that Dollar is a valid currency. Today, Bitcoins can be used to buy all sorts of goods and services online and in-person, using Bitcoin wallets, or you can exchange them for currency.

Taking into consideration some of the other changes we're seeing in our society like the decentralization of the production and distribution processes, or the way in which people are learning how to collaborate through networked technologies, bitcoins seem to be a natural way forward.

Thivyavignesh
Lakshmanaram

Alumni Interview

- Panigrahi

Bodhayan Prasad
Entrepreneur in Residence (EIR) at ZoukLoans

Year of Passing : (2005-2012)

 ❖ 2 years as an Analog Design engineer at Texas Instruments (TI)
 ❖ Young India Fellow (YIF) for a year

What would you look for while hiring new graduates?

I would look for enthusiasm, awareness, and problem solving attitude. They should think beyond their imagination and persevere to be hard working individuals.

What were your best moments in college?

It is obviously the moment when I got placed. Apart from that, getting into SPIDER was a moment of ecstasy. Completion of projects and treats were all moments to cherish.

If you had do it all again what things would you change?

"All and none ". I think my college life was perfect. I gained a lot of experience, without which, I would not have reached such heights. Friends and competitors were in perfect mixture.

What differences do you see between the present freshers and the students of your batch?

I feel everybody is distinctive. Everyone today has their own set of ideas, which are powerful ones indeed.

Tell us about a few of your goals.

"I want to build a million dollar tech company ". I would like to be a business tycoon.

Were you the part of any clubs in college? Did it help anyway?

"Being a part of SPIDER, R & D club of NIT Trichy helped me a lot. I got to work on a lot of projects. I got a good exposure towards the real corporate world, and that probably gave me enough courage to give my ideas a platform, and begin the startup "ZoukLoans"..

Apart from the academic curriculum, what interests you otherwise?

"Puzzles. I love solving them. They give your thoughts a new dimension.

Any words of advice?

"Do what your heart appeals to". Life is too short to make it monotonous by doing the conventional. Explore!

Akhila
A Pradhiksha

The story of tongue tied Nash and his table of hash.

Ayush Mishra

There was once a fellow named Nash. A perfectly jolly and happy-go-lucky person, Nash ran a Hash brown shop in the Hash brown loving city of Hashville. Quite obviously, this served him well, and he was soon a very wealthy and influential person in the city. People loved to go to his establishment which claimed to sell more than a 1000 varieties of Hash browns – all kept in a huge freezer, the size of an auditorium behind his shop. Now, it so happened that the story of Nash's famous and delicious Hash Browns reached the ears of the famous wizard Sir Dontmakemewaitalot of Impatienopolis – and so, he decided to come down to Hashville to visit Nash.This created a great hubbub in the city! This wizard was a very important person and nobody wanted to disappoint him. Nash even went ahead and hired his nephew for extra help over the weekend. The day the wizard arrived, Nash personally greeted him and took him for a city tour. They were chatting like old buddies – it was going great for Nash! Until, of course, a Hash brown was ordered.
Sir Dontmakemewaitalot, hungry after the day's activities ordered a jalapeño flavored chili cheese Hash brown, one of Nash's best sellers. Nash, as he was busy chatting with the old man, sent his nephew to fetch the crate from the freezer and cook for him. Now this is where the problem arose – Nash had been running the shop for years and had a fairly good idea about where, which supplies were stored. His nephew on the other hand, couldn't make any sense of the shelves taller than twice his height and was soon lost in the labyrinth of frozen potato patties. He did eventually emerge from the freezer room, only to face a furious wizard and a helpless Nash. As established, one simply does not make the great wizard Sir Dontmakemewaitalot, wait a lot! Within seconds, the furious wizard drew out his wand, pointed it at Nash, and said:

"You have made me wait a great deal of time
You will never speak again by the end of this rhyme
How will you sell these potato patties, grilled and fried?
When my spell will leave you helplessly tongue tied!"

And with a swish of his wand, the deed was done.

The wizard immediately left the shop, his robes flapping behind him, as Nash could only try to mumble – but all that came out of his mouth were garbled strings of gibberish. That night, the sad Uncle and his forlorn nephew just sat in the shop. His nephew, who was sure it was his fault, tried to think of a way to make it up to his uncle, and Nash – well he was just depressed and tried to read his menu as he cried "$#%^^... *#/^^... $#%^&... booohooohoohoooo!"

There must surely be a way to make it work – know what you want and where to find it in that blasted freezer at the same time.

"$#/&^... $#%&&... *#/^&... "

And then, in a moment of serendipity, it struck him! As he listened to his Uncle Nash try to read out the menu, he realized a couple of things –

1. However garbled his versions of the Hash brown names sounded, they were UNIQUE for each variety, and the same, every time he tried to say it.
2. Whatever he said, mapped to a fixed set of symbols.

So, if only he could use these names to catalogue the crates and store them, finding them would be so much easier! Now all he had to do was
segregate them by the fixed set of symbols, for example $# and *# - these symbols could be used to determine whether the hash browns he wants, are in the right half of the freezer or the left, by keeping all the Hash Browns with names beginning with $# on one side of the freezer and *# on the other. Then he could traverse each half and find exactly where to go to get his
crate using the other set of symbols his uncle blabbers out!

What Nash's nephew did here, was inadvertently discover hashing. Hashing is a very clever encoding technique that can be used for storage, encryption and what not. Every Hashing technique is centered on the hash function or our Tongue-tied Nash in this case. The unique strings generated by each value given to the Hash function, are made up of characters from a predefined domain, selected by an algorithm that makes sure these combinations assigned to each input string are unique and same every time. These strings can then be used to map these values to storage locations in data structures that store them according to these mapped characters in the string. For example, a chili cheese hash, or a "*#^//" as said by Nash would be stored on the right hand side of the freezer (*#), on the shelf labeled '^' and on the rack labeled '//'. An obvious advantage of this method would be the reduced amount of searching Nash's nephew would have to do the next time he searches for a Hash Brown. Other advantages of Hashing include:

1. Security: Only Nash would be able to tell where a particular hash brown is stored, thanks to his unique ability to produce unique garbled strings (as do most hash functions)
2. Dynamic nature: There is no need to change the cataloging every time a new item is added to the menu – all the item needs is a unique name and Nash's unique string will help his nephew determine exactly where he wants to store it
3. Reliability: It's much more reliable and flexible (than searching each crate individually, for example)

Needless to say, after the unfortunate incident, 'Nash Hash Browns' of Hashville provided the fastest and most delicious way to satisfy your hunger!

Hacker News

~ Soorya Narayan

Unjust laws exist;

shall we be content to obey them,
or shall we endeavor to amend them,
and obey them until we have
succeeded,

or shall we transgress them at once?

~ Henry David Thoreau

Recently, I stumbled upon 'The Internet's Own Boy', the story of Aaron Swartz, a tech whiz and political activist devoted to a free and open internet.

When he attempted to liberate large amounts of academic research from MIT's data network, he was faced a fine of up to $1m and 35 years in jail. Then he took his own life.

A little bit about him first. To say that he was a prodigy is an understatement. At the age of 14, Swartz helped author the RSS specification and at 19, he helped develop the social news and entertainment website Reddit, now, one of the most visited websites out there.

He also created the Markdown format and wrote a web framework called web.py. Pretty cool, huh?

Aaron wasn't just a programmer, though. Since he was a teenager he had nurtured an interest in removing any barriers to the free flow of information.

"Growing up, I slowly had this process of realising that all the things around me that people had told me about were just the natural way things were, the way things always would be, but they weren't natural at all, they were things that could be changed and more importantly, they were things that were wrong and should change. Once I realised that, there was really no going back."

In July 2011 Swartz was indicted on 13 felony counts for his massive download of JSTOR, the online pay per article archive of academic articles.

A little bit about JSTOR

JSTOR (short for Journal Storage) is a digital library founded in 1995 as a solution for managing the sheer volume of academic journals and papers.

Libraries on subscribing to JSTOR had all their records digitally stored with online access to them. This was a wonderful idea, no doubt, but the key issue here was accessibility. If your university does not have a subscription means paying anywhere between $8 to $19 to download a paper. Every year, JSTOR blocks 150 million attempts by non-subscribers to read articles. According to Harvard Law professor Lawrence Lessig, on being asked how much it would cost to make all the data free to the public, JSTOR replied with a figure of $250 million. Whoa.

What was the JSTOR incident?

A little bit of background information again. MIT operates an extraordinarily open network which is run in the spirit of the MIT ethos. At the time of Aaron's actions, the JSTOR website allowed an unlimited number of downloads by anybody on MIT's 18.x Class-A network.
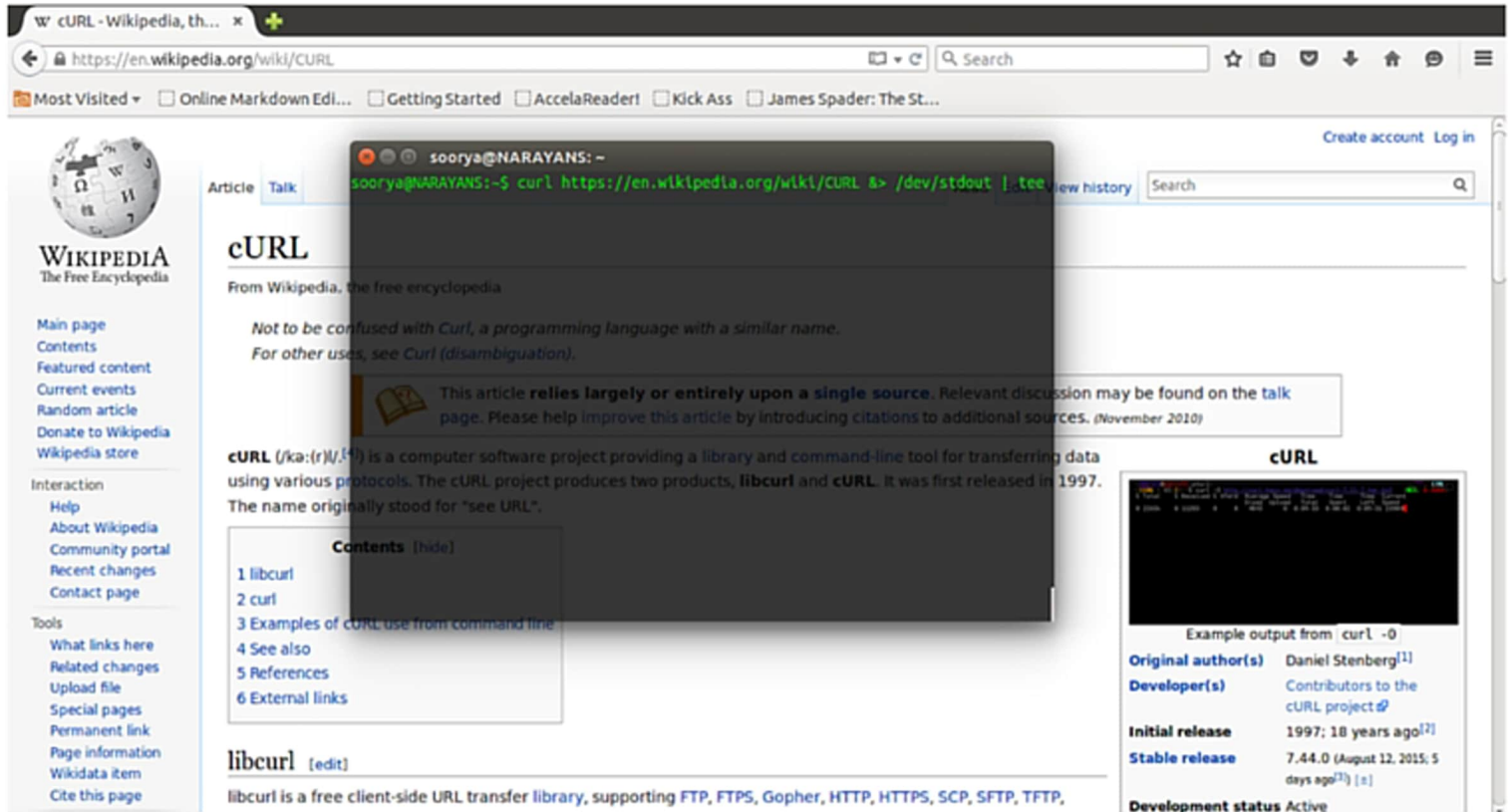
What did Aaron do?

At around September 2010, Swartz purchased a new laptop, logged into the MIT computer network, and began the liberation of JSTOR. He wrote a few basic python scripts that first discovered the URLs of journal articles and then used curl to request them. All he did was to call a basic command line tool that downloads a file in the same manner as right-clicking and choosing "Save As" from your browser. In fact, if you're running a Linux system or have installed the curl exe file in windows, you can try it out yourself. And for those of you who are too lazy to try it out, here are a couple of screen shots to show you how easy it is.

Here, I'm cURLing the Wikipedia entry on cURL and writing it into wiki.html.

Check out the URL. I'm loading the wiki.html page from my local machine here. QED.

Also, Aaron did nothing to cover his tracks or hide his activity apart from changing his computer's IP Address at regular intervals (Not a crime, mind you. If it was, you'd be arrested every time you logged into airport wifi.) Now, some of you might say, okay, he downloaded a large number of research papers, so what?

In 2008 called the "Guerrilla Open Access Manifesto." in which he says

"Providing scientific articles to those at elite universities in the First World, but not to children in the Global South? It's outrageous and unacceptable. … We need to take information, wherever it is stored, make our copies and share them with the world. … We need to download scientific journals and upload them to file sharing networks. We need to fight for Guerrilla Open Access."
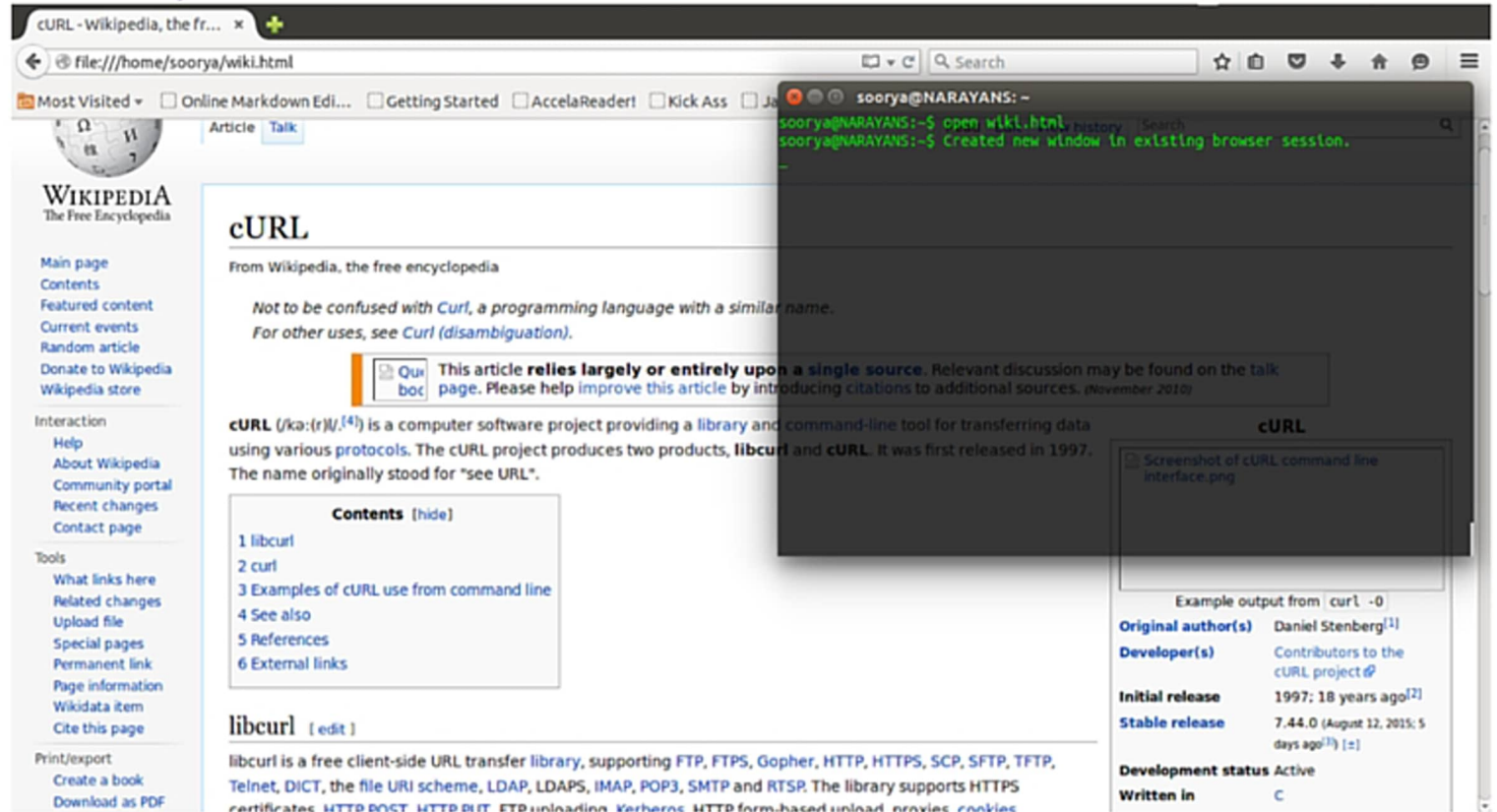
Fewer than four months later, police arrested Swartz after a surveillance camera caught him breaking into a the MIT basement, from which he was wired into the MIT network. In July of 2011, he was indicted for wire fraud, computer fraud, and recklessly damaging a protected computer. Another nine counts were added later. Swartz hanged himself in his Brooklyn apartment on January 11, 2013, while awaiting trial. What he did was not illegal and involved no 'hacking' (I mean, all he did was copy a large number of files he was supposed to have unlimited access to). Swartz unjustly became a victim of the rights and freedoms for which he stood. Swartz's death lead to many calls for more open access to scholarly data. He was posthumously inducted into the Internet Hall of Fame and was awarded the American Library Association's James Madison Award for being an "outspoken advocate for public participation in government and unrestricted access to peer-reviewed scholarly articles."

Aaron was guided by a fascination with what he saw as the corrupting influence of big money on institutions and the fundamental imbalance of power structures in the modern age. We're living in a time of great disparity. The architects of the financial meltdown are having dinner with the president and this is what the government decided to prosecute. But the question is, can we, given what's happened, make the world a better place?

Thivyavignesh
A Pradhiksha

THE FIBONACCI SERIES

-ASHUTOSH AGARWAL

Fibonacci numbers are a very interesting sequence of numbers which have large applications in searching, data structures and graphs. Essentially, fibonacci numbers are generated by the summation of the preceding two numbers, Thus

$$f(n) = f(n-1) + f(n-2)$$

Well, we all know this recursion formula and can easily use it to generate the fibo sequence in O(n) complexity, i.e, in linear time. This is good enough if we want to generate, say, the 10^8th number.

But what if we want to generate the 10^19th fibo number. Now, this problem gets interesting as, if we were to try to perform it in linear time, it would take years to compute. Well, this article is primarily to look at a genius way of computing the nth fibo number in O(logn) time. We can do this using the following generator using matrices.

$$\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n$$

Thus, to find the (n+1)th fibo number, we can take the (n)th power of the above matrix. It's well known that Power's can be computed in O(logn) using the following pseudocode:

```
Let qmatr = {initial Q-matrix}
function power (var n, mat[][] matr1) {
    if (n is 1) return qmatr;
    if (n is even) {
        matr <- power(n/2, matr1);
        return matrixMultiply(matr, matr);
    } else {
        matr <- power(n-1, matr1);
        return matrixMultiply(matr, qmatr);
    }
}
```

This algorithm can do the computations in O(logn) but if we look at the constant involved, we notice that at each step, matrix multiplication has to be performed which is O(n^3). At best, we can perform this using Strassen's Algorithm in O(n^2.8). For a 2X2 matrix multiplication operation, this is approximately O(6.96) in constant time.

Thus, the net complexity comes to about 6.96*O(logn).

At this point, things get more interesting, as it turns out, we can use another property of Fibonacci Numbers to do this entire operation in just 4*O(logn).

The property is as follows:

```
If F(n) is the (n+1)th fibo number,
and F(0) = F(1) = 1
then,
if (n is even) {
    k=n/2;
    F(n) = F(k)*F(k) + F(k-1)*F(k-1);
} else {
    n = 2*k + 1;
    F(n) = F(k)*F(k-1) + F(k)*F(k+1);
}
```

At some point of time, the linear computation approach was probably thought to be the only way possible, but the above algorithms just show that the complexity of the problem is only as hard as our understanding of it.

It would be immensely silly to assume that we have ever reached the best possible solution to a problem.

AKHILA YERUKOLA
SHILPA SWETH

Startup

Comparison of Uber, Lyft And OLA

- Arvindmani Satyanarayan

An extremely fashionable start up, Uber inspired many other On-Demand cab services around the world. The company develops, markets and operates the Uber app, which allows consumers with a smart phone to submit a trip request which is then routed to Uber drivers who use their own cars. Since the success of the transportation behemoth, several other companies have mimicked its business model, a trend that has come to be referred as **'Uberification'**.

By mid 2015, Uber was estimated to be worth $50b. However, they face many legal challenges all over the world by governments and other companies alike, who allege its use of unlicensed drivers is unsafe and illegal.

Its more modest cousin, **Lyft**, is another startup that was founded around the same time. Also an on-demand cab service, Lyft has a focus on ridesharing by connecting passengers who need a ride with drivers who have a car.

Amidst the recent controversies related to the drivers hired, one tenet of Lyft's platform is establishing trust among its users. All drivers undergo a strict screening process, and after each ride the passengers and drivers can rate each other, establishing both their reputations within the network.

Uberification wasn't limited by national boundaries, and crept into India with the Bangalore based startup Ola, which quickly emerged as a strong presence in the country. Founded in 2010, the company has been valued at $5b as of September 2015. Spanning an ever growing expansive network, the service grew to incorporate autos during December 2014. Ola supports both cash and cashless payment in the form of Ola money, and claims to clock more than 150,000 bookings per day and command 60% of market share in India.

Go !

ThousandEyes

ThousandEyes is a lnetwork monitoring company that produce software with the same name that analyses the performance of local and wide area networks. The product, which is marketed as a Software-as-a-Service (SaaS), uses synthetic monitoring probes to measure performance.

In synthetic monitoring, also known as active monitoring, behavioural scripts are created to simulate an action or path that a customer or an end-user might take on that site. These paths are then continuously monitored at specific intervals for performance.

The company was founded in 2010 by Mohit Lad and Ricardo Oliveira, and the next year received a $500k National Science Foundation grant to study DNS infrastructure. Only later in June 2013 did they launch their network monitoring product, it has been mentioned in many lists of top start-ups, including Forbes'.

Zenefits

Another company with a SaaS product, Zenefits was named the fastest growing company in Silicon Valley in 2015. Their cloud based product manages the human resources for companies, with a focus on helping them with health usage coverage.

Zenefits was started by Parker Conrad, the current CEO, and Laks Srini, Conrad's colleague at an earlier job, to help start-ups and small businesses find insurance quotes and manage employee benefits in one place.

In January 2015, Zenefits announced that it had revenue of approximately US$20 million in 2014, twenty times the corresponding figure in 2013, despite offering its platform to users for free and making all its money from commissions charged to insurers for being the insurance brokers.

Parker Conrad once revoked an offer to an employee who posted a question on Quora to choose between Uber and Zenefits, wondering which job would help him land a job at Google in the future, which led to both criticisms and praises.It has been pointed out that the company's extreme pace of growth has put a lot of pressure on Zenefits, with even Conrad saying that the problems that other companies have a year to solve need to be solved by them within eight weeks.

Thivyavignesh
A Pradhiksha

DIGITAL PEN

-RAHUL KAR

Tired of important documents going into hiding right when you need them? Do you often find yourself writing down contact numbers on a small piece of paper instead of storing it on your smartphone, and then, cursing yourself when you can't find them? Behold the Digital Pen, a new technology invented by the Swedish inventor and entrepreneur Christer Fåhraues.
So, how exactly does the Digital Pen help?
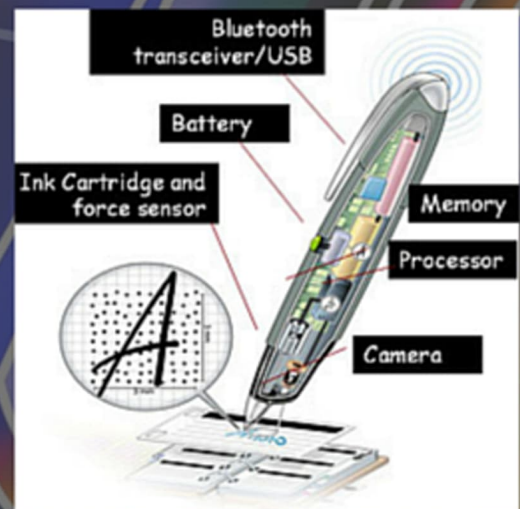
The Digital Pen is an input device which captures the handwriting or brush strokes of a user, converting handwritten analog information created using "pen and paper" into digital data, thereby enabling the data to be utilized in various applications. For example, the written data can be digitized and uploaded onto a computer and displayed on the monitor. The data can then be interpreted by handwriting softwares (OCR) and be used in different applications or just as graphics. Digital pens typically contain internal electronics and have features such as touch sensitivity, input buttons, memory, writing data transmission capabilities and electronic erasers.

The computer pen is comparable to a regular ink pen (and even uses refillable ink!) that writes on regular paper, except that it has an optical reader that records motion, images and coordinates. You can browse and edit your written notes, diagrams, tables or even doodles! Another useful feature of this computer invention is that hand-written digital files can be easily converted into text fonts for use in word documents or emails.

The Digital Pen comes in many varieties:

- ## Accelerometer-based-pens

  Accelerometer-based digital pens contain components that detect movement of the pen and contact with the writing surface.

- ## Active-pens

  Active pens, such as N-trig's Duo Sense Pen, include electronic components whose signals are picked up by a mobile device's built-in digitizer and transmitted to its controller, providing data on pen location, pressure, button presses and other functionality.

- ## Position-based-pens

  Position-based digital pens use a facility to detect the location of the tip during writing. Some models can be found on graphic tablets made popular by Wacom, and on tablet computers using Wacom's Penabled technology.

- ## Camera-based-pens

  Camera-based pens use special digital paper to detect where the stylus contacts the writing surface, such as those using Anoto technology.

- ## Trackball pen

  Trackball pens use sensors to detect the motion of the trackball.

This fancy invention has been licensed to companies around the world for various commercial products. Applications include data/signature capture, completing forms, mapping, surveying, document management, paper replay, whiteboards, toys and education.

It's okay to forget, now that the Digital Pen is around.

ANUGRAHAA RAMESH
SHILPA SWETH

Legendary Programmers

Linus Torvalds: An Inspirational story

Parth Thakkar

:~$ **L**inus Torvalds, well known for the creation of the Linux kernel, is an extremely interesting figure. He has received numerous awards, and is extremely well known within the community of computer programmers. He is also the creator of the wildly popular distributed revision control system - **git.**

:~$ **R**ise to the fame

:~$ **L**inus graduated from the University of Helsinki in 1996, with a master's degree in Computer Science. During this time, he bought a book that changed his life (and the rest of ours too) – Andrew Tanenbaum's Operating Systems: Design and Implementation, where Tanenbaum describes MINIX, an educational, stripped-down version of UNIX. Impressed by the clear structure of UNIX and its underlying philosophy, he started working on Linux, in 1991. The first prototypes of Linux were publicly released later that year. It was a rather small, hobby project for him. In early 1992, about six months after announcing the creation of Linux, Torvalds posted an online message asking anyone using the operating system to send him a postcard.

:~$ **S**oon, his mailbox in Helsinki overflowed with hundreds of postcards from the United States, New Zealand, Japan, and beyond. It was the first time that his sister and mother, with whom Torvalds shared an apartment, realized that he was up to something big. Torvalds had told them little about what he was doing in his bedroom, perched over his computer, all hours of the day and night. The Linux kernel was on its way to become world's largest collaborative development project.

:~$ **F**rom 1997 to 1999, he was involved in 86open helping to choose the standard binary format for Linux and Unix. In 1999, he was named by the MIT Technology Review TR100 as one of the world's top 100 innovators under age 35.

:~$ **I**n 1999, Red Hat and VA Linux, both leading developers of Linux based software, presented Torvalds with stock options in gratitude for his creation. That same year, both companies went public and Torvalds' share value temporarily shot up to roughly US$20 million.

:~$ **C**urrently, the Linux Foundation sponsors Torvalds so he can work full-time on improving Linux.

:~$ Controversies and differences

:~$ **L**inus is well known not just for the software he has created, but also for his extremely frank manner of speaking. Profanity and insults have long been management tactics of Linux creator Linus Torvalds. Many have complained about how he's turned the open source development world into a hostile environment to work in. One of the developers has called him "one of the worst offenders".

:~$ **B**esides this, he also has a lot of philosophical differences with many, many people. There has been a growing concern regarding security issues in Linux kernel. For Linus, security bugs are just like any other bugs. For many others, especially for the likes of people managing security at NSA, nuclear power plants, stock exchanges – all powered typically by the Linux kernel – this is a huge issue. Many people have tried pushing for improving the security of the kernel, however, Linus – the ultimate authority to decide which new code is incorporated into the standard Linux kernel – has fundamental differences with such people. For him, the most important thing is performance. If he thinks that introducing some code reduces performance "unreasonably", then that code is not worthy enough to be included in the kernel, even if it is a security bug-fix.

:~$ The humane side

:~$ **A**ll this might suggest that he is an asocial, computer nerd – which he has always claimed to be. However, Torvalds turns out to be an easy conversationalist with a puckish, self-deprecating sense of humour.

:~$ **W**hile Linus Torvalds is a cult figure among computer enthusiasts worldwide, he's essentially invisible outside the tech world, and prefers a quiet life. After several years in the Silicon Valley, Torvalds and his family, tired of the region's suburban anonymity, decided to move to Oregon – a place that is at a more human pace, where people are more interested in neighbours and friends.

:~$ **I**n an interview, he said "I actually like having stuff nearby, even though I never go to it. I like having a small downtown, and I like knowing that I could, if I wanted to, do things. That makes me happy."

:~$ **A**nd then," Torvalds continued, "I can ignore it and do my own thing anyway."

Anugrahaa V R
Lakshmanaram

Bits & Bytes now welcomes freelancers to send in their work to be included in the subsequent editions.