Parallel Algorithms Come of Age

Guy Blelloch
Professor and Associate Dean for UG programs

Merriam-Webster

Definition of ALGORITHM

: a procedure for solving a mathematical problem (as of finding the greatest common divisor) in a finite number of steps that frequently involves repetition of an operation; *broadly* : a step-by-step procedure for solving a problem or accomplishing some end especially by a computer

- a search *algorithm*

History of Algorithms

Erastosthenes (200 BC)

al-Khwarizmi (800 AD)

FFT (Gauss 1805)

Min-Spanning-Tree (Boruvka 1926)

Effective computability (Church/Turing 1935-37)

History of Algorithms

Dijkstra's shortest path algorithm (1956)

Quicksort (1959)

Complexity Theory (Hartmanis/Stearns/Blum 1965)

Knuth volumes 1-3 (1968)

NP completeness (Cook-Levin 1971)

big-O (Hopcroft/Tarjan 1972)

The Design and Analysis of Computer Algorithms (Aho/Hopcroft/Ullman 1974)

Cost measured in terms of sequential time/number of steps.

Algorithms at Universities

Reequired in All CS Undergraduate Programs, e.g.

CMU : 15-210 (..Algorithms), 15-451 (Algorithms)

MIT : 6.006 (Introduction to Algorithms)
6.046 (Design and Analysis of Algorithms)

Berkeley: 61B (Data Structures), 170 (Efficient Algorithms)

Stanford: 161 (Data Structures and Algorithms)

…

Algorithms in the "Real World"

sign in | become a supporter | subscribe | search | jobs | dating | more | UK edition

theguardian

UK | world | politics | sport | football | opinion | culture | business | lifestyle | fashion | environment | tech | travel | browse all sections

home › science

Mathematics

How algorithms rule the world

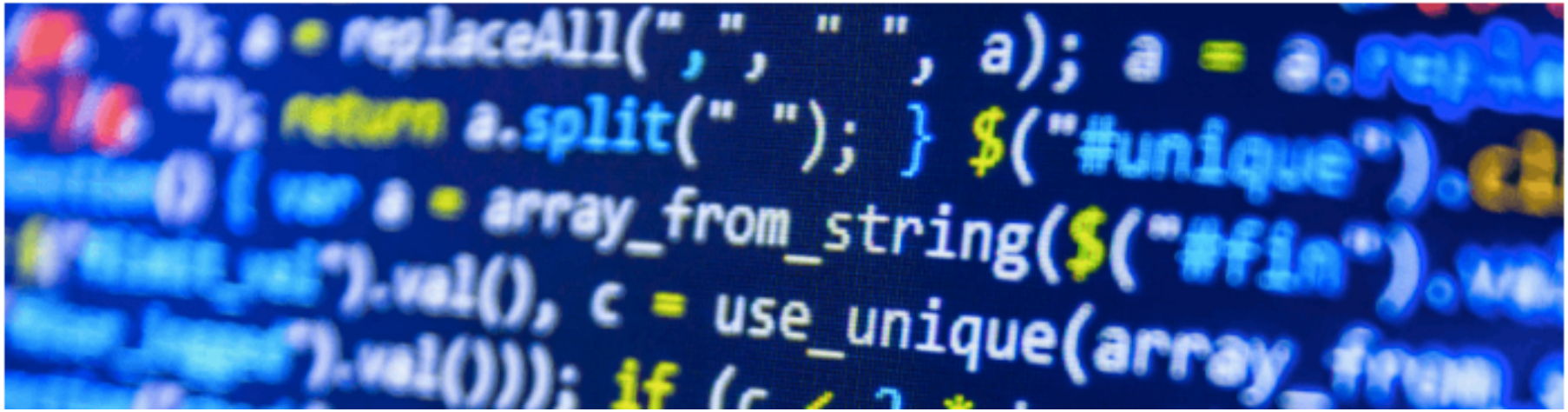
The NSA revelations highlight the role sophisticated algorithms play in sifting through masses of data. But more surprising is their widespread use in our everyday lives. So should we be more wary of their power?

New algorithms may revolutionize drug discoveries—and our understanding of life

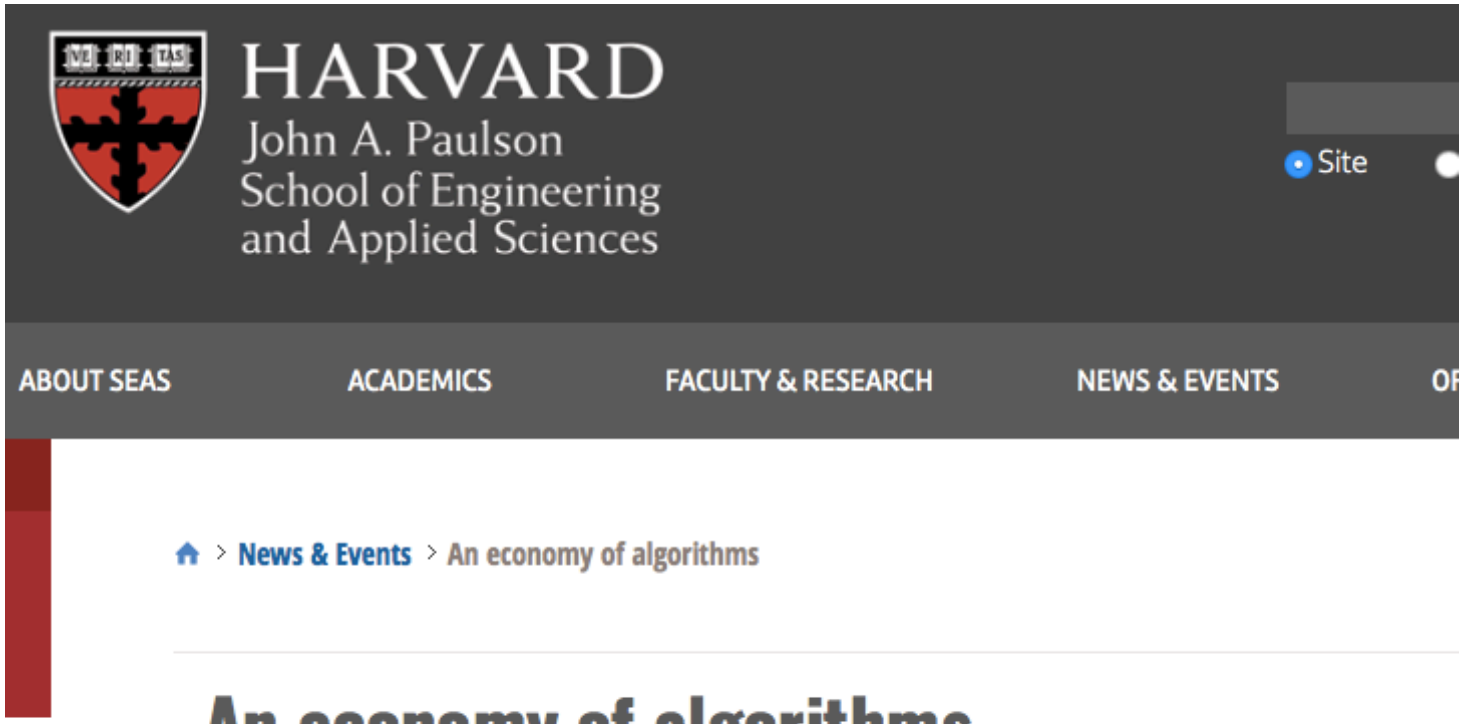
February 6, 2017

Why Future Emphasis Should be on Algorithms – Not Code

By **Linda Johnson** - March 26, 2017



Trend in Tech


HARVARD
John A. Paulson
School of Engineering
and Applied Sciences
Site
ABOUT SEAS
ACADEMICS
FACULTY & RESEARCH
NEWS & EVENTS
News & Events
An economy of algorithms
An economy of algorithms
ComputeFest 2017 tackles the future of the computational economy
By Leah Burrows | January 27, 2017

CITYFIXER

Using Algorithms To Predict Gentrification

Data analysts are trying to give community development advocates the tools they need to fight displacement and economic decline.

TANVI MISRA | @Tanvim | Feb 17, 2017 | 2 Comments

Why San Francisco courtrooms are turning to computer algorithms for advice

By HOLLY MCDEDE • JAN 26, 2017

PROGRAM
Crosscurrents

Tweet
Share
Google+
Email

MARGARET RHODES DESIGN 10.03.16 2:39 PM

SO. ALGORITHMS ARE DESIGNING CHAIRS NOW

PROCESS AUTOMATION DESK

By Aaron Hand , Executive Editor, on July 7, 2016

FOLLOW FOLLOW

Big Data Algorithms Optimize Oil Wells

Ambyint has added a new product to a lineup geared toward taking the analysis out of data analytics, reducing labor costs associated with operating oil wells.

MIT Technology Review

Topics+

Business Impact

Algorithms Probably Caused a Flash Crash of the British Pound

Trading software may have overreacted to tweets about the French president's comments on Brexit.

by Jamie Condliffe October 7, 2016

Why are (Sequential) Algorithms so Successful?

1. Theory exactly predicts runtimes?
2. Are good for highly performing optimized codes?
3. Will impress our friends?

Maybe

Why are Sequential Algorithms so Successful?

1. Teaching abstraction
2. Good for explaining core ideas, and why they are useful
3. Well defined and simple cost model which is "good enough" for asymptotic comparisons
4. Simple pseudocode and small step to real code that can be easily compiled and run to get reasonably efficient code.
5. Sequential algorithms are elegant

What about "Parallel Algorithms"

Wikipedia: a **parallel algorithm** is an algorithm which can be executed a piece at a time on many different processing devices, and then combined together again at the end to get the correct result.[1]

History of Parallel Algorithms

Matrix Inversion (1977)
Merging: Valiant (1978)
PRAM Model (1980)
NC Algorithms (1983)
Log depth sorting (1985)
Many efficient algorithms (82-90)
Jaja: Parallel Algorithms (1990)

-- Mid 1990s, hibernation of parallel computing

Why Parallel Algorithms Now?

1. Almost all computers and devices are now parallel
2. Many applications, from small to large, need the performance
3. Simple programming + cost models
4. There exist efficient parallel algorithms for most problems in theory and
5. Availability of reasonable tools and languages
6. **Many sequential algorithms are already parallel.**

Not an advanced topic

Why Parallelism: Machines

Intel Xeon Eight-Core E5-2660 2.2GHz 8.0GT/s 20MB LGA2011 Processor without Fan, Retail BX80621E52660

by Intel

Be the first to review this item

Price: $137.99 & **FREE Shipping**

Get $40.00 off instantly: Your cost could be **$97.99** upon approval for the Amazon.com Store Card. Learn more

Note: Not eligible for Amazon Prime. Available with free Prime shipping from other sellers on Amazon.

Only 13 left in stock.

Get it as fast as Thursday, Oct. 13.

Ships from and sold by Galactics.

- Model: Intel Xeon Processor E5-2660
- Core Count: 8
- Clock Speed: 2.2 GHz
- Cache: 20 MB
- Max Memory Bandwidth: 51.2 GB/s
- Socket: LGA2011

Used & new (29) from $47.95 + $6.44 shipping

Why Parallelism: Machines

INTEL XEON 20 CORE PROCESSOR E5-2698V4 2.2GHZ 50MB SMART CACHE 9.6 GT/S QPI TDP 135W

by Intel

1 customer review

Price: $3,599.95 + $4.27 shipping

Your cost could be **$3,594.95**: Qualified customers get $5 in Gift Card funds on first $100 reload of their Amazon Gift Card Balance. Learn more

Note: Not eligible for Amazon Prime.

Only 6 left in stock.

Want it Thursday, Oct. 13? Order within **5 hrs 16 mins** and choose **Two-Day Shipping** at checkout. Details

Ships from and sold by Mega Micro Devices Inc..

- # of Cores 20
- # of Threads 40
- Processor Base Frequency 2.2 GHz
- Max Turbo Frequency 3.6 GHz
- NEW OEM - 1 year warranty

New (6) from $3,129.71 + $4.51 shipping

64 core blade servers ($6K) (shared memory)

amazon.com

Hello. Sign in to get personalized recommendations. New customer? Start here.

Your Amazon.com | Today's Deals | Gifts & Wish Lists | Gift Cards

Shop All Departments | Search | Electronics

Computers & Accessories | Brands | Best Sellers | Laptops, Tablets & Netbooks | Desktops & Servers | Computer Accessories & Peripherals | Computer Parts &

Amd Opteron (sixteen-core) Model 6274

by AMD

Be the first to review this item | Like (0)

Price: $792.99

In Stock.

Ships from and sold by J-Electronics.

Only 1 left in stock--order soon.

4 new from $714.03

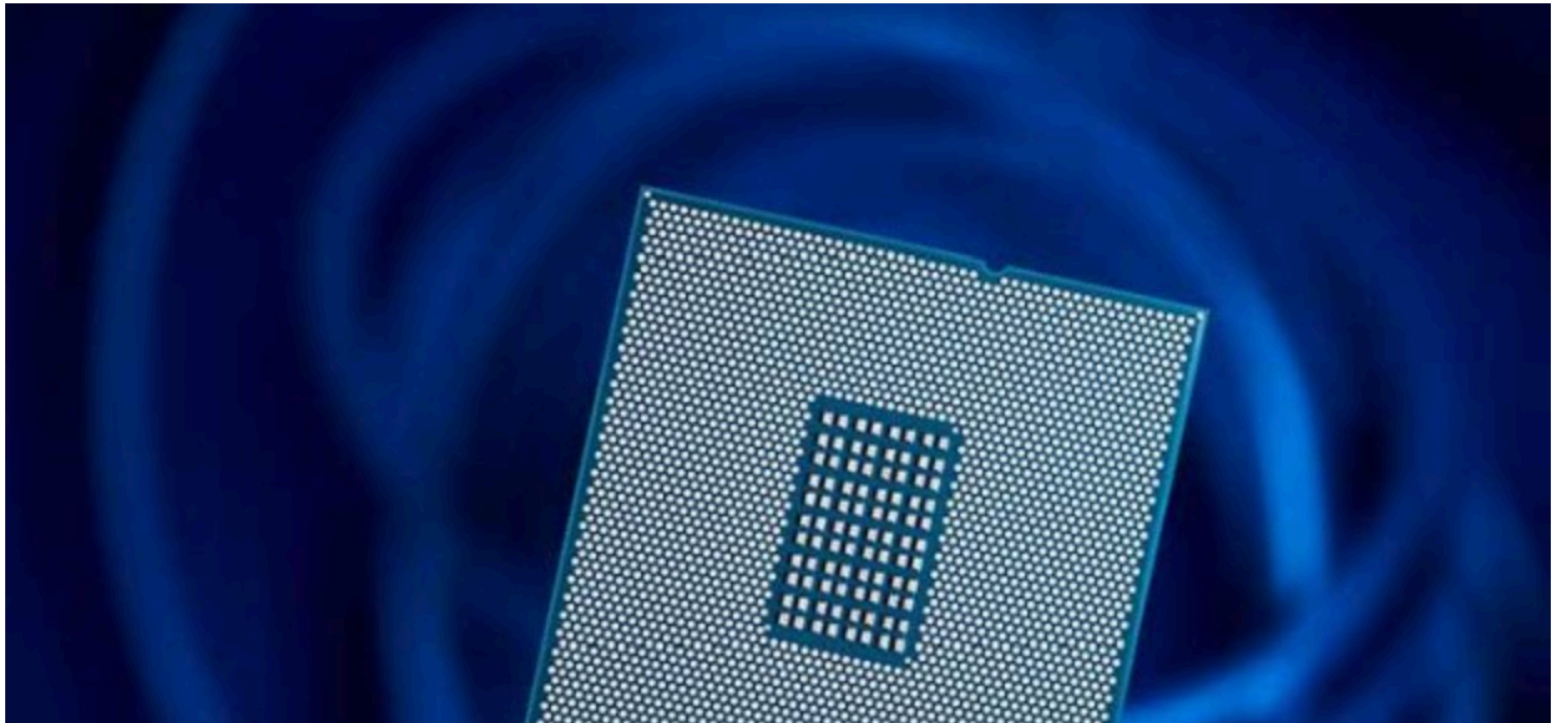
x 4 =

Xeon Phi: Knights Landing (64 cores)

Qualcomm readies up 48-core Centriq 2400 ARM server chip

by Zak Killian — 12:51 PM on December 9, 2016

Maybe 2017 will be the year that ARM servers finally become a thing. After demoing a 24-core server chip a little more than a year ago, Qualcomm's Datacenter Technologies subsidiary has announced the Centriq 2400 CPU. This new chip is a 48-core ARMv8 processor based on a new in-house CPU core design called Falkor, and it's compliant with ARM's Server Base System Architecture specification. Earlier in the week, Qualcomm showed off the new hardware running "a typical datacenter application" comprising Linux with Java and Apache Spark.



4992 "cuda" cores

Roll over image to zoom in

Nvidia Tesla K80 24GB GPU Accelerator passive cooling 2x Kepler GK210 900-22080-0000-000

by NVIDIA

29 customer reviews | 11 answered questions

Price: $4,295.95 + $11.55 shipping

Note: Not eligible for Amazon Prime.

In Stock.

Ships from and sold by eServer PRO.

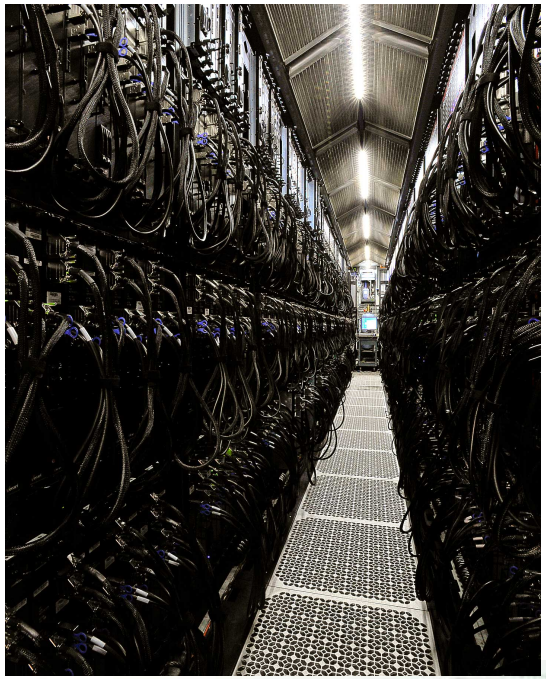
Estimated Delivery Date: Aug. 27 - Sept. 1 when you choose Expedited at checkout.

- Nvidia Tesla K80 GPU: 2x Kepler GK210
- Memory size (GDDR5) : 24GB (12GB per GPU)
- CUDA cores: 4992 (2496 per GPU)
- Memory bandwidth: 480 GB/sec (240 GB/sec per GPU)
- 2.91 Tflops double precision performance with NVIDIA GPU Boost - See more at: http://www.nvidia.com/object/tesla-servers.html#sthash.IF5LVwFq.dpuf

4 new from $4,135.00

Upgrading to a Solid-State Drive?

Learn how to install an SSD with Amazon Tech Shorts. Learn more



Up to 300K servers

LG Optimus 2X: first dual-core smartphone launches with Android, 4-inch display, 1080p video recording

Chris Ziegler
12.15.10

Dec 2010

53
Shares

Samsung Galaxy S IV to feature Exynos 28nm quad-core processor?

Written by Andre Yoskowitz @ 01 Nov 2012 18:02

Nov 2012

It has been a few weeks but there is a new rumor regarding the upcoming Samsung Galaxy S IV.

According to reports, Samsung will pack next year's flagship device with its "Adonis" Exynos processor, a quad-core ARM 15 beast that uses efficient 28nm tech.

Samsung is supposedly still testing the application processor, but mass production is scheduled for the Q1 2013 barring any delays.

SEP 4, 2014 @ 09:07 AM 11,224 VIEWS

Lenovo Announces First Octa-Core Smartphone, The Vibe X2

Jay McGregor, CONTRIBUTOR

I cover all aspects of technology and enterprise. FULL BIO

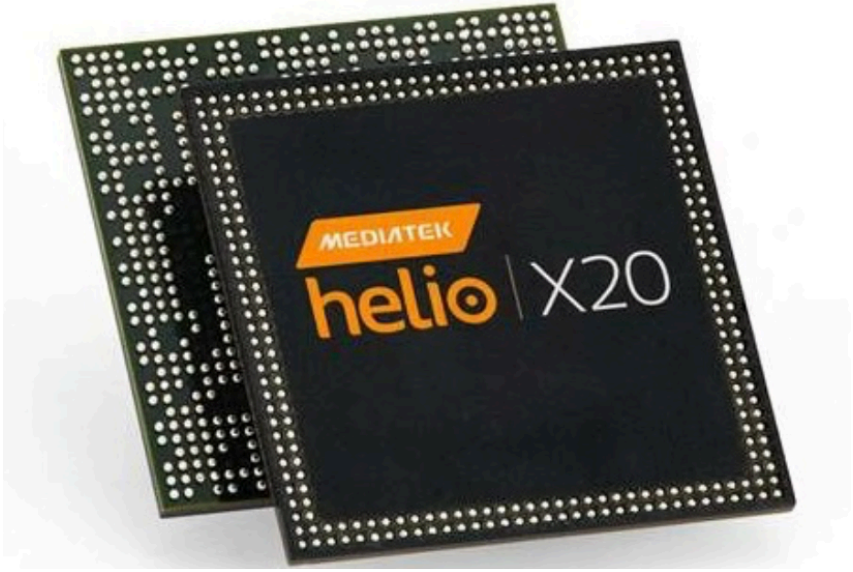
Opinions expressed by Forbes Contributors are their own.

Sep 2014

10-core MediaTek Helio X20 is official

NEWS  by **Robert Triggs** ▪ May 12, 2015

May 2015



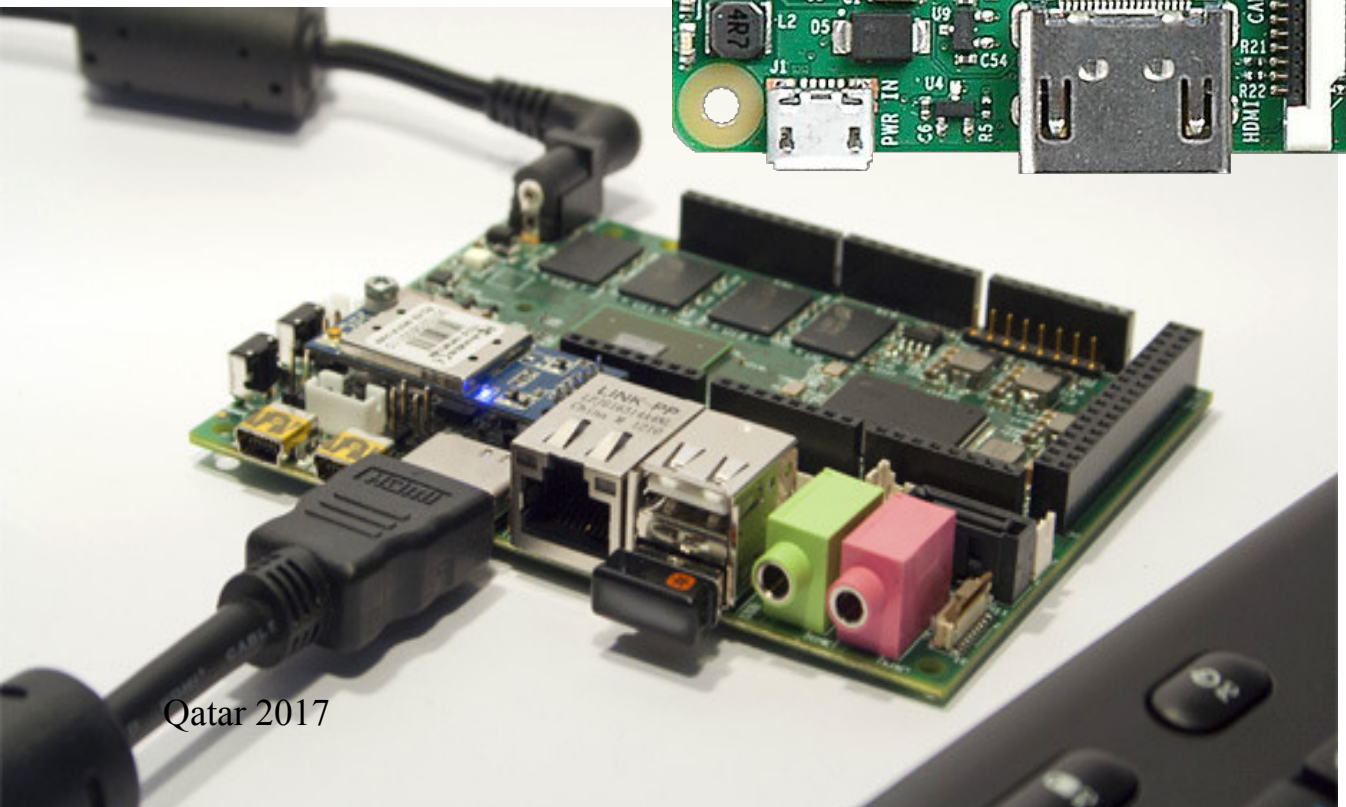
MediaTek and TSMC trialing new 7nm smartphon processor with mad CPU core count

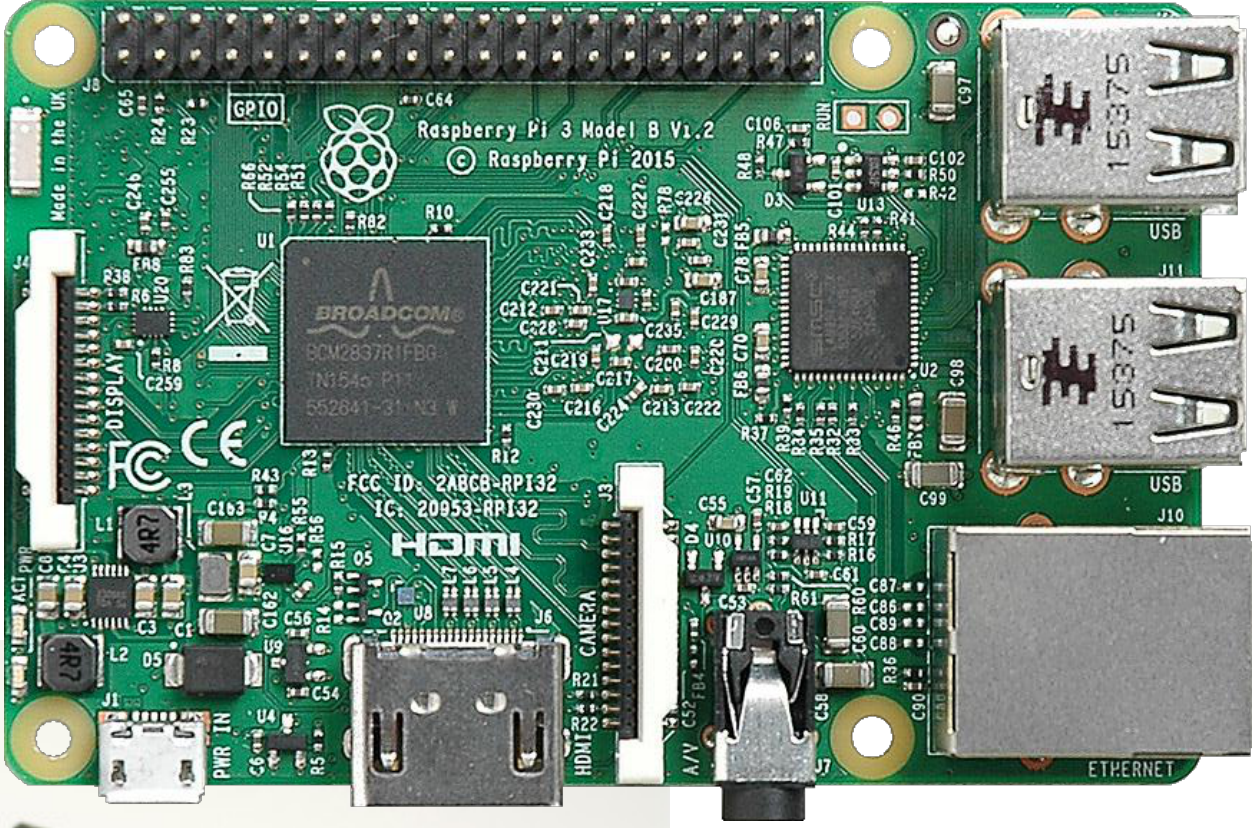
Posted: 09 Mar 2017, 06:53, by Luis D.

Share Tweet Share

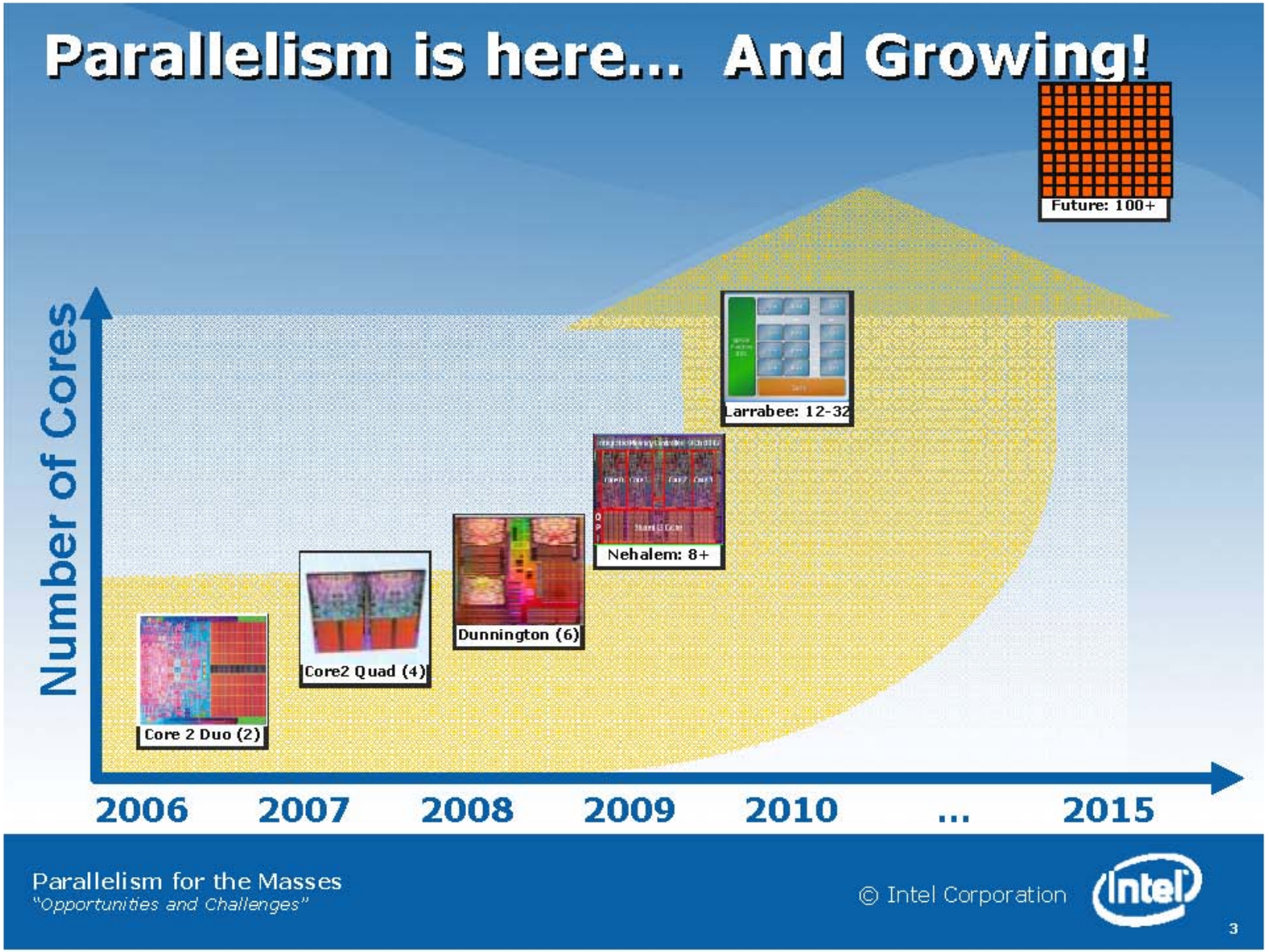
Mar 2017, 12 core

UDOO





Raspberry Pi 3


Parallelism is here... And Growing!
Future: 100+
Larrabee: 12-32
Nehalem: 8+
Dunnington (6)
Core2 Quad (4)
Core 2 Duo (2)
Number of Cores
2006
2007
2008
2009
2010
...
2015
Parallelism for the Masses
"Opportunities and Challenges"
© Intel Corporation
intel
3

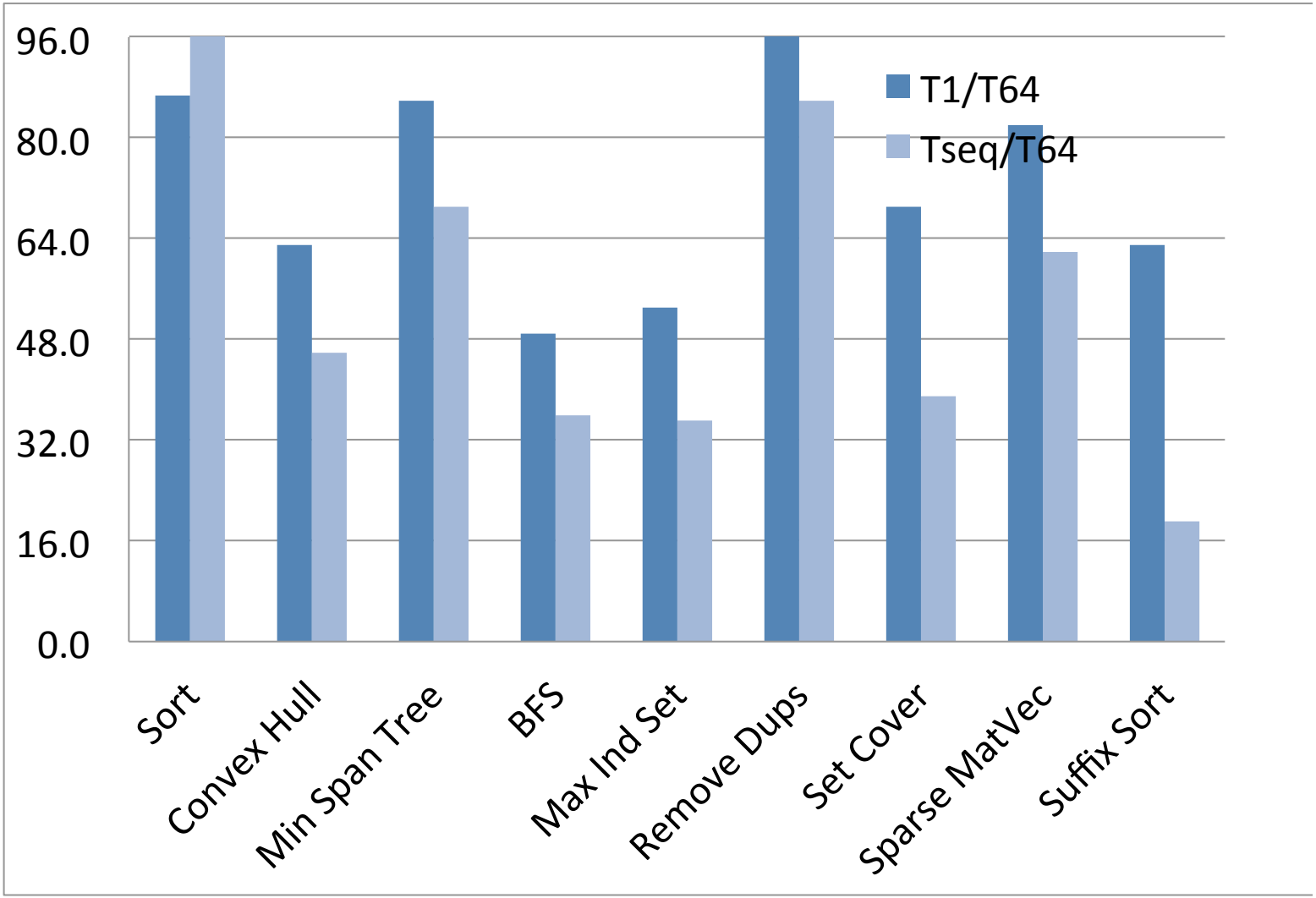
Why Parallel Algs.: Applications

Big : Weather prediction, airplane design, genome sequencing, oil exploration, molecular modeling, sky survey analysis, social network analysis, traffic modeling, epistomology, route-scheduling, delivery optimization, business analytics, high-speed trading, online advertising, …

Small : signal processing, graphics, encryption, compression, AI, vision, language understanding, virtual reality, computer games, search, self-driving-cars

Most mentioned earlier are actually parallel

Why Parallel Algs.: Practical Efficiency



64 core Xeon Phi

Why Parallel Algorithms: Theoretical Efficiency

Cost in terms of:

- **Work:** total number of operation
- **Span:** longest chain of dependences

Ratio gives **parallelism**

Examples:

- Sorting, O(n log n), O(log n)
- Merging O(n), O(log n)
- Spanning Trees O(m), O(log n)
- …

Why Parallel Algs.: Programming

parallel loops

```
cilk_for (i=0; i < n; i++)
    B[i] = A[i]+1;
```

Cilk

```
Parallel.ForEach(A, x => x+1);
```

Microsoft TPL (C#,F#)

```
B = {x + 1 : x in A}
```

Nesl, Parallel Haskell

```
#pragma omp for
for (i=0; i < n; i++)
   B[i] = A[i] + 1;
```

OpenMP

Why Parallel Algs.: Programming

fork join

```
cobegin { S1; S2;}
```

Dates back to the 60s. Used in dialects of Algol, Pascal

```
coinvoke(f1,f2)
```

Java fork-join framework

```
cilk_spawn S1;
S2;
cilk_sync;
```

Cilk

```
(e1 || e2)
```

SML (as used in 210)

Why Parallel Algs.: Programming

Tools

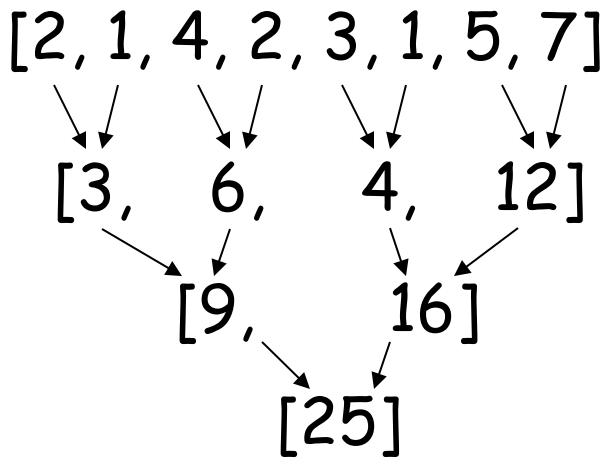
- Debuggers
- Race detectors
- Profiling

What does a Parallel Algorithm Look like

It is not rocket science

except when applied to rocket science

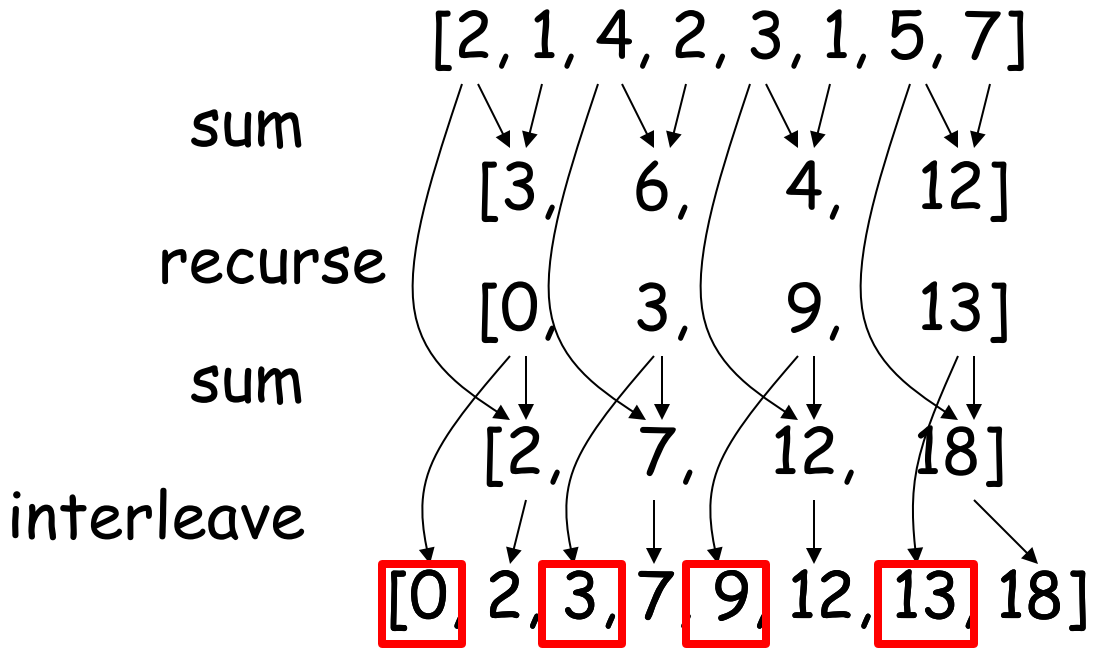
Example: summing an array




```
function sum(A) =
if (#A <= 1) then [0]
else sum({A[2*i] + A[2*i+1] : i in [0:#a/2]})
```

W(n) = W(n/2) + O(n) = O(n)
D(n) = D(n/2) + O(1) = O(log n)

Example: scan



Scan code

```
function addscan(A) =
if (#A <= 1) then [0]
else let
  sums = {A[2*i] + A[2*i+1] : i in [0:#a/2]};
  evens = addscan(sums);
  odds = {evens[i] + A[2*i] : i in [0:#a/2]};
in interleave(evens,odds);
```

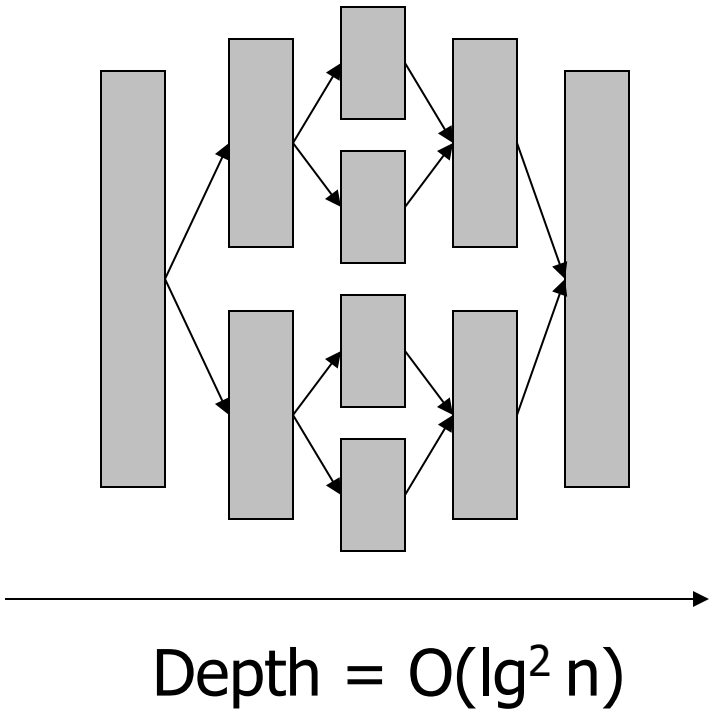
W(n) = W(n/2) + O(n) = O(n)
D(n) = D(n/2) + O(1) = O(log n)

Example: Quicksort

```
function quicksort(S) =
if (#S <= 1) then S
else let
  a = S[rand(#S)];
  S1 = {e in S | e < a};
  S2 = {e in S | e = a};
  S3 = {e in S | e > a};
  R = {quicksort(v) : v in [S1, S3]};
in R[0] ++ S2 ++ R[1];
```

Quicksort (nested parallelism)

Analyze in terms or Work (W) and Depth (D)



Depth = $O(\lg^2 n)$

Work = $O(n \lg n)$

Parallelism = $W/D = O(n/ \lg n)$

Time = $W/P + D$

P = # processors

Why Parallel Algs.:Techniques

Some common themes in "Thinking Parallel"

1. Working with collections.
   - map, selection, reduce, scan, collect
2. Divide-and-conquer
   - Even more important than sequentially
   - Merging, matrix multiply, FFT, …
3. Contraction
   - Solve single smaller problem
   - List ranking, graph contraction
4. Randomization
   - Symmetry breaking and random sampling

Why Parallel Algorithms Now?

1. Almost all computers and devices are now parallel
2. Many applications, from small to large, need the performance
3. Simple programming + cost models
4. There exist efficient parallel algorithms for most problems in theory, and cool techniques
5. Availability of reasonable tools and languages
6. **Many sequential algorithms are already parallel.**

Not an advanced topic

Algorithms in Education

Reequired in All CS Undergraduate Programs, e.g.

CMU : 15-210 (..Algorithms), 15-451 (Algorithms)

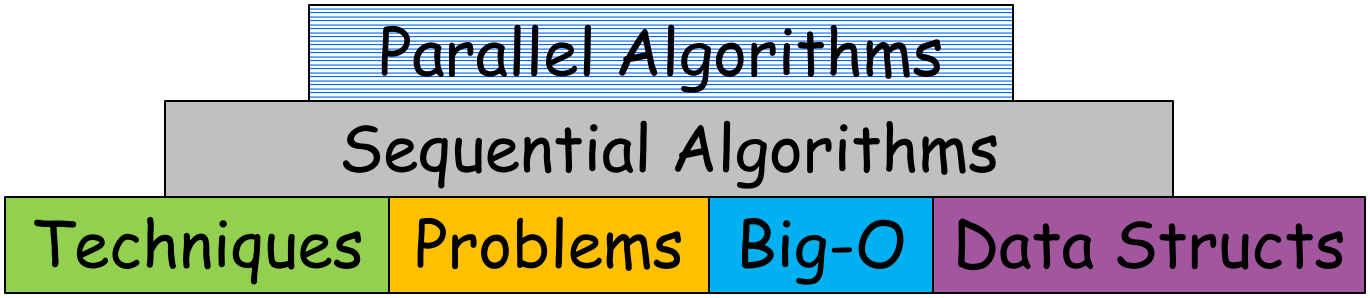
MIT : 6.006 (Introduction to Algorithms)
6.046 (Design and Analysis of Algorithms)

Berkeley: 61B (Data Structures), 170 (Efficient Algorithms)

Stanford: 161 (Data Structures and Algorithms)

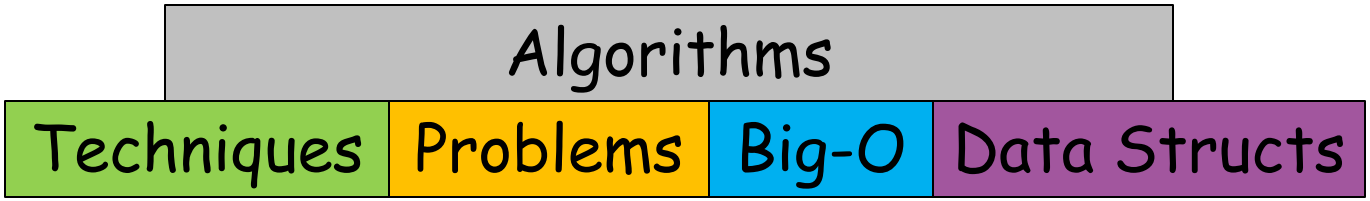
…

Beyond 210, very little parallel algorithms.



Wrong way to think about it:

Instead:



Conclusions

Algorithms are very important

Many reasons to use parallel algorithms

Happened so rapidly that a large part of the world has not yet caught on

Needs to be better integrated with undergraduate curriculums