

# EECS 252 Graduate Computer Architecture

## Lec 21 – Goodbye to Computer Architecture, Bad Career Alternatives, & Cal Cultural History

David Patterson  
Electrical Engineering and Computer Sciences  
University of California, Berkeley

<http://www.eecs.berkeley.edu/~pattsrn>  
<http://vlsi.cs.berkeley.edu/cs252-s06>

1

## Outline

- Goodbye to Computer Architecture
- Part IV: Key Advice on Alternatives to a Bad Career
- (Post-Tenure) Career Advice from Richard Hamming, Ivan Sutherland, Karl Pister
- Your Cal Cultural Heritage
- Course Evaluations
- Return projects, final grades

2

## Chapter 1: Introduction

- CS 252 to learn new skills, transition to research
- Computer Science at the crossroads from sequential to parallel computing
  - Salvation requires innovation in many fields, including computer architecture
- Computer Architecture skill sets are different
  - Technology tracking and anticipation
  - Solid interfaces that really work
  - Quantitative approach to design
  - 5 Quantitative principles of design:
    1. Take Advantage of Parallelism
    2. Principle of Locality
    3. Focus on the Common Case
    4. Amdahl's Law
    5. The Processor Performance Equation
- Computer Architecture >> instruction sets

3

## Chapter 2: Dynamic Execution Processor

- Leverage Implicit Parallelism for Performance: Instruction Level Parallelism
- Loop unrolling by compiler to increase ILP
- Branch prediction to increase ILP
- Dynamic HW exploiting ILP
  - Works when can't know dependence at compile time
  - Can hide L1 cache misses
  - Code for one machine runs well on another
- Reservations stations: *renaming* to larger set of registers + buffering source operands
  - Prevents registers as bottleneck
  - Avoids WAR, WAW hazards
  - Allows loop unrolling in HW
- Not limited to basic blocks (integer units gets ahead, beyond branches)
- Helps cache misses as well
- Lasting Contributions
  - Dynamic scheduling, Register renaming, Load/store disambiguation

4

## Chapter 3: Static Instruction Level Parallelism

- Limits to ILP (power efficiency, compilers, dependencies ...) seem to limit to 3 to 6 issue for practical options
- Explicitly parallel (Data level parallelism or Thread level parallelism) is next step to performance
- Coarse grain vs. Fine grained multithreading
  - Only on big stall vs. every clock cycle
- Simultaneous Multithreading if fine grained multithreading based on OOO superscalar microarchitecture
  - Instead of replicating registers, reuse rename registers
- Itanium/EPIC/VLIW is not a breakthrough in ILP
- Balance of ILP and TLP decided in marketplace

5

## Appendix H: Vector Architecture

- Vector is alternative model for exploiting ILP
- If code is vectorizable, then simpler hardware, more energy efficient, and better real-time model than Out-of-order machines
- Design issues include number of lanes, number of functional units, number of vector registers, length of vector registers, exception handling, conditional operations
- Fundamental design issue is memory bandwidth
  - With virtual address translation and caching

6

## Goodbye to Processors & Questions for 2011

- Did vector ever become popular for multimedia and/or signal processing?
- Did EPIC/VLIW die?
- Given switch the power limits and the switch to multiprocessors, did microarchitecture complexity
  1. Continue to increase ( >6 issue, >6 complete)
  2. Freeze as of ~ 2002 (e.g., Power 4, Opteron)?
  3. or Regress? (e.g., Niagara)?
- Did fine grain multithreading help with efficiency and/or did it make the parallel programming harder since scale of parallelism is larger and declined in popularity?

7

## Chapter 4: Multiprocessors

- Caches contain all information on state of cached memory blocks
- Snooping cache over shared medium for smaller MP by invalidating other cached copies on write
- Sharing cached data  $\Rightarrow$  Coherence (values returned by a read), Consistency (when a written value will be returned by a read)
- Snooping and Directory Protocols similar; bus makes snooping easier because of broadcast (snooping  $\Rightarrow$  uniform memory access)
- Directory has extra data structure to keep track of state of all cache blocks
- Distributing directory  $\Rightarrow$  Scalable shared address multiprocessor  $\Rightarrow$  Cache coherent, Non uniform memory access

8

## Goodbye to MultiProcessors & Questions for 2011

- Did processors / chip continue 2X every 2 yrs?
  - Desktop vs. servers vs. embedded
  - How many sockets can easily put together?
- Any innovations in synchronization or communication taking advantage of cores on the same chip?
- Did everything remain cache coherent all the time, or can turn off and use just message passing?
- Any changes to the ISA to make it easier to support parallel programming? (e.g., Transactional memory?)
- Did we need new languages/compiler to use them? Threads vs. messages? MPI still popular?
- Did enhancements to performance accountability/predictability simplify parallel programming?
- Performance/Power only, or SPUR too?
- What % of peak did PETAFLIPS machine deliver?

9

## Chapter 5: Advanced Memory Hierarchy

- Memory wall inspires optimizations since so much performance lost there
  - Reducing hit time: Small and simple caches, Way prediction, Trace caches
  - Increasing cache bandwidth: Pipelined caches, Multibanked caches, Nonblocking caches
  - Reducing Miss Penalty: Critical word first, Merging write buffers
  - Reducing Miss Rate: Compiler optimizations
  - Reducing miss penalty or miss rate via parallelism: Hardware prefetching, Compiler prefetching
- “Auto-tuners” search replacing static compilation to explore optimization space?
- DRAM – Continuing Bandwidth innovations: Fast page mode, Synchronous, Double Data Rate

10

## Chapter 5: Advanced Memory Hierarchy

- Virtual Machine Revival
  - Overcome security flaws of modern OSes
  - Processor performance no longer highest priority
  - Manage software, manage hardware
- Virtualization challenges for processor, virtual memory, I/O
  - Paravirtualization, ISA upgrades to cope with those difficulties
  - Xen as example VMM using paravirtualization

11

## Goodbye to Memory Hierarchy & Questions for 2011

- What was maximum number of levels of cache on chip? Off chip? (Did it go beyond 3?)
- Did local memory (IBM Cell) replace/enhance caches?
- Did DRAM latency improve? How much better DRAM BW? Rate of improvement in capacity? 2X / 3 years?
- Did Virtual Machines become very popular? Which was primary reason?
  - Enhance Security, Manage Hardware, Manage Software?
- Is software shipped inside a VM?
- Is x86 now as low overhead for VMM now as IBM 370? How many iterations did it take?
- Compared to 2006, are security/privacy/break-ins
  - 1. Worse; 2. Same; 3. Better?
- If 3, Did architecture/hardware play an important role?

12

## Chapter 6: Advanced Storage

- Disks: Aerial Density now 30%/yr vs. 100%/yr in 2000s
- TPC: performance/\$ to normalize configurations
  - Auditing to ensure no foul play
  - Throughput with restricted response time is normal measure
- Fault  $\Rightarrow$  Latent errors in system  $\Rightarrow$  Failure in service
- Components often fail slowly
- Real systems: problems in maintenance, operation as well as hardware, software
- Queuing models assume state of equilibrium: input rate = output rate
- Little's Law:  $Length_{system} = rate \times Time_{system}$   
(Mean number customers = arrival rate x mean service time)
- Appreciation of latency vs. utilization
- Clusters for storage as well as computation
- RAID: reliability  $\gg$  performance for storage

13

## Goodbye to Storage & Questions for 2011

- How did disk capacity improve? Seek time? BW?
- Are disks basically unchallenged for storage, or did flash or NVDRAM make major inroads?
- TPC-C still used for benchmark?
- Did SATA, SAS, FC-AL interfaces all remain popular?
- Is RAID 6 the norm? RAID-7? (recover 3 errors)
- Did tapes finally die, replaced with remote disks?
- Which became the most popular model of distributed (storage/caching) services:
  1. Model A (Akamai): 1000s of data centers, each with dozens of computers
  2. Model G (Google): dozens of data centers, each with 1000s of computers
- Did Statistical Machine Learning (RAD Lab) significantly help storage/distributed systems?

14

## 252 as introduction to research

- Did “research-in-the-small” in CS252
- Excellent results as evidenced in poster session and presentations, papers
  - Lots of nice comments to me on selection of topics, quality of the work
  - Included many people from outside Berkeley
- Some work will be carried forward this summer in RAMP project
- If interested in doing more, talk to me

15

## Outline

- Goodbye to Computer Architecture
- Part IV: Key Advice on Alternatives to a Bad Career
- (Post-Tenure) Career Advice from Richard Hamming, Ivan Sutherland, Karl Pister
- Your Cal Cultural Heritage
- Course Evaluations
- Return projects, final grades

16

## One Alternative Strategy to a Bad Career

- Caveats:
  - From a project leader's point of view
  - Works for me; not the only way
  - Primarily from academic, computer systems perspective
- Goal is to have impact:  
Change way people do Computer Science & Engineering
  - Academics have bad benchmarks: number published papers
- 6 Steps
  - 1) Selecting a problem
  - 2) Picking a solution
  - 3) Running a project
  - 4) Finishing a project
  - 5) Quantitative Evaluation
  - 6) Transferring Technology

17

## 1) Selecting a Problem



### Invent a new field & stick to it?

- No! Do "Real Stuff": solve problem that someone cares about
  - Follow the problem vs. the fad
- No! Use separate, short projects
  - Always takes longer than expected
  - Matches student lifetimes
  - Long effort in fast changing field??
  - Learning: Number of projects vs. calendar time
  - If going to fail, better to know soon
- Strive for multi-disciplinary, multiple investigator projects
  - 1 expert/area is ideal (no arguments)
- Match the strengths and weaknesses of local environment
- Make sure you are excited enough to work on it for 3-5 years
  - Prototypes help

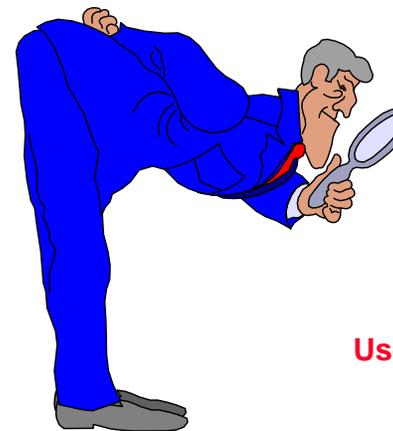
18

## My first project

- Multiprocessor project with 3 hardware faculty ("Xtree")
- 1977: Design our own instruction set, microprocessor, interconnection topology, routing, boards, systems, operating system
- Unblemished Experience:
  - none in VLSI
  - none in microprocessors
  - none in networking
  - none in operating systems
- Unblemished Resources:
  - No staff
  - No dedicated computer (used department PDP-11/70)
  - No CAD tools
  - No applications
  - No funding
- Results: 2 journal papers, 12 conference papers, 20 TRs
- Impact?

19

## 2) Picking a solution



### Let Complexity Be Your Guide?

- No! Keep things simple unless a very good reason not to
  - Pick innovation points carefully, and be compatible everywhere else
  - Best results are obvious in retrospect "Anyone could have thought of that"
- Complexity cost is in longer design, construction, test, and debug
  - Fast changing field + delays  
⇒ less impressive results

### Use the Computer Scientific Method?

- No! Run experiments to discover real problems
- Use intuition to ask questions, not to answer them

20

(And Pick A Good Name!)

**R**educed  
**I**nstruction  
**S**et  
**C**omputers

**R**edundant  
**A**rray of  
**I**nexpensive  
**D**isks

**R**esearch  
**A**ccelerator for  
**M**ultiple  
**P**rocessors

...

21

### 3) Running a project

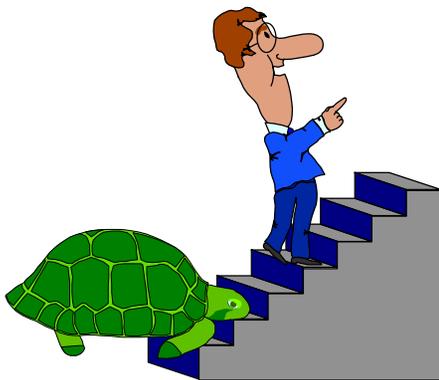


#### Avoid Feedback?

- No! Periodic Project Reviews with Outsiders
  - Twice a year: 3-day retreat
  - faculty, students, staff + guests
  - Key piece is feedback at end
  - Helps create deadlines, team spirit
  - Give students chance to give many talks, interact with others industry
- Consider mid-course correction
  - Fast changing field & 3-5 year projects ⇒ assumptions changed
- Pick size and members of team carefully
  - Tough personalities are hard for everyone
  - Again, 1 faculty per area reduces chance of disagreement

22

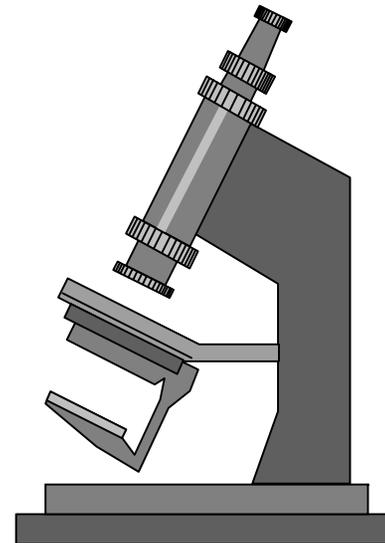
### 4) Finishing a project



- People count projects you finish, not the ones you start
- Successful projects go through an unglamorous, hard phase
- Design is more fun than making it work
  - “No winners on a losing team; no losers on a winning team.”
  - “You can quickly tell whether or not the authors have ever built something and made it work.”
- Reduce the project if its late
  - “Adding people to a late project makes it later.”
- Finishing a project is how people acquire taste in selecting good problems, finding simple solutions

23

### 5) Evaluating Quantitatively



#### Never be Proven Wrong?

- No! If you can't be proven wrong, then you can't prove you're right
- Report in sufficient detail for others to reproduce results
  - Can't convince others if they can't get same results
- For better or for worse, benchmarks shape a field
- Good ones accelerate progress
  - Good target for development
- Bad benchmarks hurt progress
  - Help real users v. help sales?

24

## 6) Transferring



### Publishing Journal Papers IS Technology Transfer?

- **No! Missionary work: “Sermons” first, then they read papers**
  - Selecting problem is key: “Real stuff”
    - » Ideally, more interest as time passes
    - » Change minds with believable numbers
    - » Prima Donnas interfere with transfer
- **My experience: industry is reluctant to embrace change**
  - Howard Aiken, circa 1950:
    - “The problem in this business isn’t to keep people from stealing your ideas; its **making** them steal your ideas!”
  - Need 1 bold company (often not no. 1) to take chance **and** be successful
    - » RISC with Sun, RAID with (Compaq, EMC, ...)
  - Then rest of industry must follow

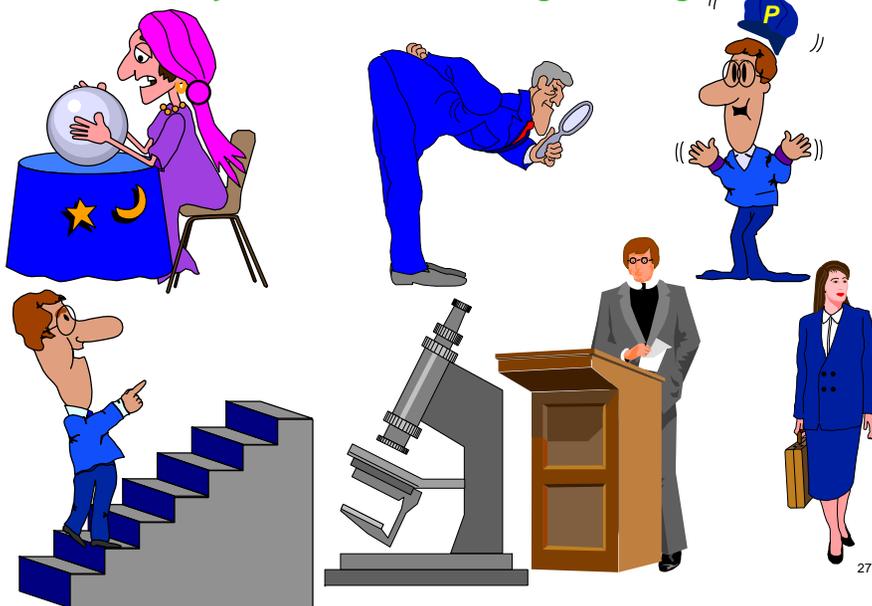
25

## 6) Transferring Technology



- **Pros**
  - Personal satisfaction: seeing your product used by others
  - Fame (people remember companies)
  - Learn a lot
  - Personal \$\$\$ (potentially)
- **Cons**
  - Only 10% of startups really make it
    - » Fame if company unsuccessful too
  - Learn about business plans, sales vs. marketing, financing, personnel benefits, hiring, firing, ...
    - » Spend time doing above vs. research/development<sup>6</sup>

### Summary: Leader’s Role Changes during Project



27

### Richard Hamming’s Advice: “You and Your Research” (Later in your Research Career)

- **Doing Nobel Quality Research**
  - Search Google for transcript of 1986 talk at Bell Labs
- **Luck?** “Luck favors the prepared mind.” Pasteur
- **Important Problems:** “Great Thoughts Time” Friday afternoons
- **Courage:** think about important, unsolved problems
  - Big results usually to problems not recognized as such, and people usually did not get encouragement
- **Working conditions:** can use creatively to lead to original solutions
  - Bell labs didn’t have acres of programmers
- **Drive:** what distinguishes the great scientists
  - Not brains; commitment vs. dabbling; compound interest over time
- **Open doors** (vs. closed offices): short term vs. long term benefit
- **Selling the work:** not only published, but people must read it
  - as much work spent on polish and presentation as on the work itself
- **Age:** After 1st big success, hard to work on small problems
  - So change field at least every 10 years
- **Educate your boss, Stimulation, right amount of Library work**

28

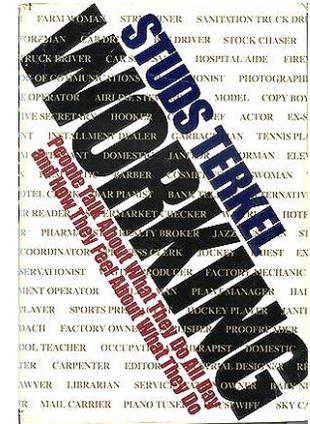
## Ivan Sutherland's Advice: "Technology And Courage"

- **Courage:** to perceive risk and proceed in spite of it
  - Technology: risks to reputation and pride
  - Research: high probability that an attempt will fail
  - If inadequate courage, work up courage, reduce risk, reduce perception of risk, or don't do it
- 1. **External Encouragement** (rewards and punishment)
  - Deadlines, groups of people, mentors, seminars, tenure, taking / teaching classes, starting companies, stock
- 2. **Self Encouragement**
  - Getting started: warm-up project, break into tasks and do 1<sup>st</sup> one
  - To continue: refuse to let urgent drive out the important
- 3. **Rewards**
  - Thrill of discovery, following curiosity, beauty, simplicity
  - Personal joy of playing with technology
- <http://research.sun.com/techrep/Perspectives>

29

## Teaching and Students, Family and Friends for Career Satisfaction

- When deciding on my career while finishing Ph.D., read the book "Working" by Studs Terkel
  - Oral history of working life
  - My reaction was 2 groups of jobs led to good feelings about careers
1. People who designed or built things that lasted (e.g., Golden Gate Bridge, Empire State Building): architects, construction workers, ...
  2. People who helped or trained people: teachers, nurses, doctors ...
- Also, importance of family and friends in addition to careers
  - People with jobs who built transient products that have long faded left less good about careers: aerospace engineers, ...

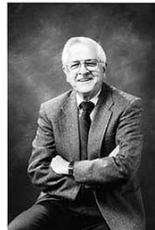


30

## Importance of Students, Family, Friends

Karl Pister

Alumnus of the Year, UC Berkeley, 2006  
Member, National Academy of Engineering  
Chancellor, UC Santa Cruz  
Dean, College of Engineering, UC Berkeley  
Father of 6  
Grandfather of 10



- ... when asked about his greatest engineering accomplishments, he demurs, saying:  
"I have always believed that the development of people is more important than developing things. I take more pride in my students than in the research papers we produced."

"Alumnus of the Year Karl Pister - A man of the campus,"  
By Patrick Dillon, *California*, January 2006

31

## Annual Patterson Family Reunion

- 1 week every 4<sup>th</sup> of July in Santa Cruz, CA
- 22 people + 3 dogs
- Play, talk, reconnect
- I'm the tour director
- See more:  
[davidpatterson.com/PattersonReunion2005/](http://davidpatterson.com/PattersonReunion2005/)



32

## Old Friends!

- I play soccer every Sunday with same group since 1980



- I go away to play poker all weekend twice a year with old high school classmates



## Acknowledgments

- Many of these ideas were borrowed from (inspired by?) Tom Anderson, David Culler, Al Davis, Ken Goldberg, John Hennessy, Steve Johnson, John Ousterhout, Randy Katz, Bob Sproull, Carlo Séquin, Bill Tetzlaff, and many others

## Conclusion: Alternatives to a Bad Career

- Goal is to have impact:  
**Change way people do Computer Science & Engineering**
  - Many 3 - 5 year projects gives more chances for impact
- Feedback is key: seek out and value critics
- Do “**Real Stuff**”: make sure you are solving some problem that someone cares about
- Taste is critical in selecting research problems, solutions, experiments, and communicating results;
  - Taste acquired by feedback and completing projects
- **Faculty real legacy is people, not papers:**
  - Create environments that develop professionals of whom you are proud
    - **Students** are the coin of the academic realm
- **Family and old friends** are the coin of the real world

## What to Emphasize about Cal culture?

- **Top public university** for undergraduate education? (US News)
- **Top graduate program** in the world?
  - 35/36 departments in top 10
- **2<sup>nd</sup> best university** in the world
  - 2004 *Times Higher Education Supplement*
- **Faculty Awards?**
  - 6 current Nobel Prize winners (18 all time)
  - 28 current “Genius” awards (MacArthur fellows)
  - 3 current Fields Medalists
  - 87 in National Academy of Engineering
  - 127 in National Academy of Science
  - Source: <http://www.berkeley.edu/about/honors/>

## Cal Cultural History: ABCs of Football

- Started with “soccer” (aka football)
  - 11 on a team, 2 teams, 1 ball, on a field; object is to move ball into “goal”; most goals wins. No hands!
- New World changes rules to increase scoring:
  - Make goal bigger! (full width of field)
  - Carry ball with hands
  - Can toss ball to another player backwards or laterally (called a “lateral”) anytime and forwards (“pass”) sometimes
- How to stop players carrying the ball? Grab them & knock them down by making knee hit the ground (“tackle”)
  - In soccer tackle the ball; football tackle the person

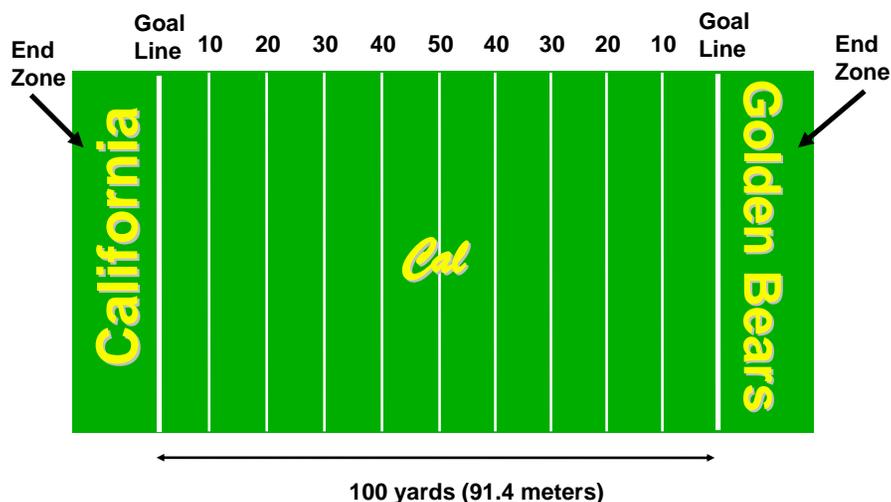
37

## ABCs of American Football

- Score by...
  - Moving football into goal (“cross the goal line” or “into the end zone”) scoring a “touchdown” (6 points)
  - Kicking football between 2 poles (“goal posts”) scoring a “field goal” ( worth 3 points, unless after touchdown, then its just 1 point: “extra point” )
- Kick ball to other team after score (“kickoff”)
  - laterals OK
- Game ends when no time left (four 15 min quarters) and person with ball is stopped
  - Soccer: two 45 min halves, time stops play

38

### Football Field



39

### The Spectacle of American Football

- Cal’s archrival is **Stanford**
  - Stereotype is Private, Elitist, Snobs
  - E.g, Manchester City vs. Manchester United
- Play nearby archrival for last game of season
  - Called “**The Big Game**”: Cal vs. Stanford, winner gets a trophy (“The Axe”) : Oldest rivalry west of Mississippi; 100th in 1997
- American college football is a spectacle
  - School colors (Cal **Blue** & **Gold** v. **Red** & **White**)
  - Nicknames (Golden Bears v. Stanford Cardinal)
  - School mascot (Oski the bear v. a tree(!))
  - Leaders of cheers (“cheerleaders”)

40

## The Spectacle of American Football

- “Bands” (orchestras that march) from both schools at games
- March & Play
  - Before game, at halftime, after game
- Stanford Band more like a drinking club; (Seen the movie “Animal House”?)
  - Plays one song: “All Right Now”
  - Cannot march and play

41

## 2005 Big Game: Cal 27 to 3 over Stanford



Cal's 4th consecutive big game victory

42

## 1982 Big Game: “The Play”

“Top 20 favorite sports event in 20th century”,  
*Sports Illustrated*

“The Greatest Display of Teamwork in the History of Sport”  
*Several sportswriters*

“...The Play, widely considered the most dramatic ending in college football history”, AP news

“...widely considered the most famous play in college football history,” *Stanford Magazine*

(“The Play” Has own entry in Wikipedia)

- **Stanford**
  - Quarterback is John Elway, who goes on to be a professional All Star football player (retired 1999)
  - Possibly greatest quarterback in college history?
    - » In 1982, they had lost 4 games in last minutes
- Stanford has just taken lead with 4 seconds left in game; Cal team captain yells in huddle “Don’t fall with the ball!”;  
[watch video](#)

43

## Notes About “The Play” (1/3)



“Allright here we go with the kick-off. Harmon will probably try to squib it and he does. Ball comes loose and the Bears have to get out of bounds. Rogers along the sideline, another one... they're still in deep trouble at midfield, they tried to do a couple of....the ball is still loose as they get it to Rogers. They get it back to the 30, they're down to the 20...Oh the band is out on the field!! He's gonna go into the endzone!!! He got into the endzone!! ...

THE BEARS HAVE WON!!! THE BEARS HAVE WON!!! Oh my God, the most amazing, sensational, dramatic, heart rending... exciting thrilling finish in the history of college football!”

– KGO's Joe Starkey

44

## Notes About “The Play” (2/3)

- Cal only had 10 men on the field; last second another came on (170 pound Steve Dunn #3) and makes key 1st block
- **Kevin Moen #26**: 6’1” 190 lb. safety,
  - laterals to Rodgers (and doesn’t give up)
- **Richard Rodgers #5**: 6’ 200 lb. safety, Cal captain “Don’t fall with that ball.”
  - laterals to Garner
- **Dwight Garner #43**: 5’9” 185 lb. running back
  - almost tackled, 2 legs & 1 arm pinned, laterals
- **Richard Rodgers #5 (again)**: “Give me the ball!”
  - laterals to Ford

45

## Notes About “The Play” (3/3)

- **Mariet Ford #1**: 5’9”, 165 pound wide receiver
  - Smallest player, leg cramps; overhead blind lateral to Moen **and** blocks 3 players
- **Moen** (again) cuts through Stanford band into end zone (touchdown!), smashes Trombonist
- On field for Stanford: 22 football players, 3 Axe committee members, 3 cheerleaders, 144 Stanford band members (172 for Stanford v. 11 for Cal)
  - “**Weakest part of the Stanford defense was the woodwinds.**” -- Cal Fan
- Cal players + Stanford Trombonist (Gary Tyrrell) hold reunions; Stanford revises history (20-19 on Axe) and still hasn’t accepted loss

46

## Thanks to Archana Ganapathi as TA

- My busiest semester in a while
- Thanks for Archana for lots of help!

47

## The Future for Future Cal Alumni

- What’s The Future?
- New Century, Many New Opportunities: Parallelism, Statistics + CS, Bio + CS, Society (Health Care, 3<sup>rd</sup> world) + CS
- Cal heritage as future alumni
  - Hard Working / Can do attitude
  - Never Give Up (“Don’t fall with the ball!”)
- **“The best way to predict the future is to invent it”** – Alan Kay (inventor of personal computing vision)

**Future is up to you!**

48