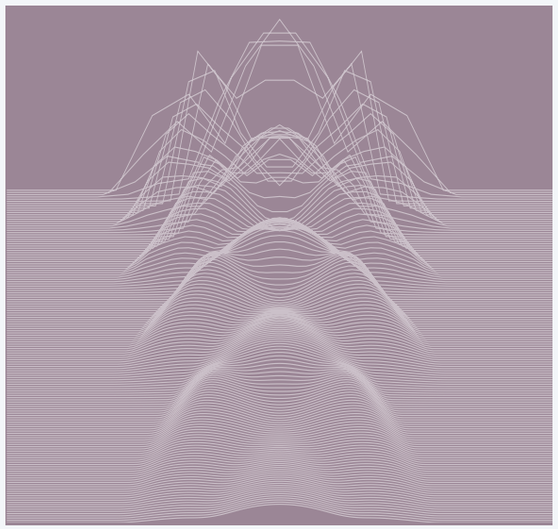
A 21st Century Model for Disseminating Knowledge

Robert Sedgewick
Princeton University

[joint work with Kevin Wayne]

A 21st Century Model for Disseminating Knowledge

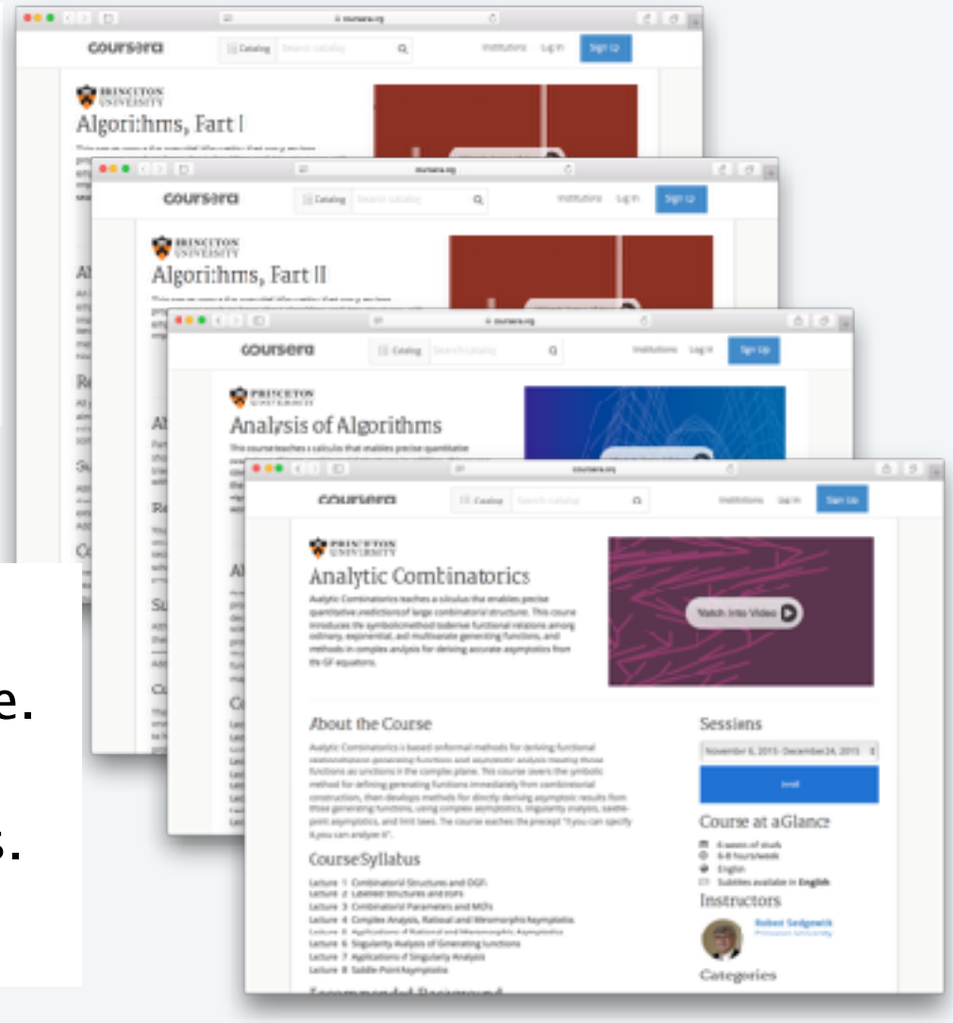


- Mission accomplished?
- Taking the plunge
- A way forward
- Postscript

Brief summary of MOOC experience

Facts and figures

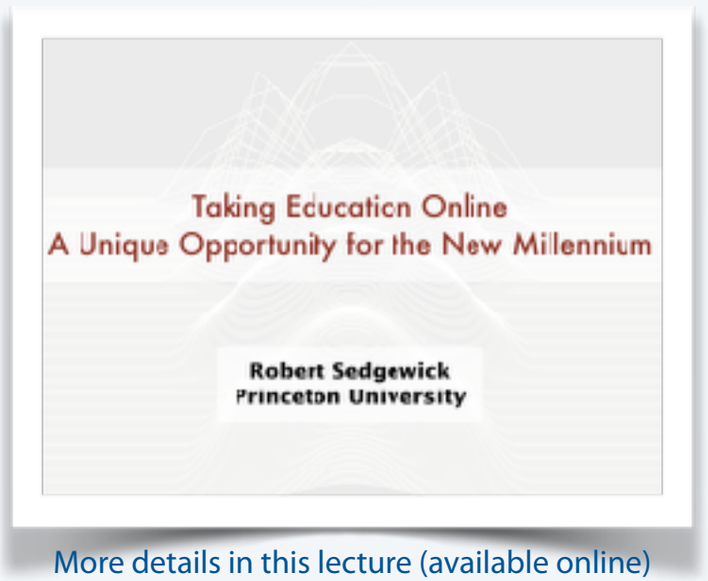
- Four courses produced and deployed.
- 45+ lectures, each running 60-90 minutes.
- 2000+ state-of-the-art lecture slides.
- Over 1 million people reached.



Distribution model is evolving (stay tuned)

- Coursera MOOCs completed in 2013, still active.
- Each course has an associated textbook.
- Lecture videos also bundled with the textbooks.
- Each textbook has associated web content.

Q: What's been happening since 2013?

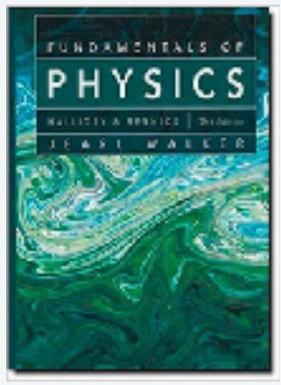


More details in this lecture (available online)

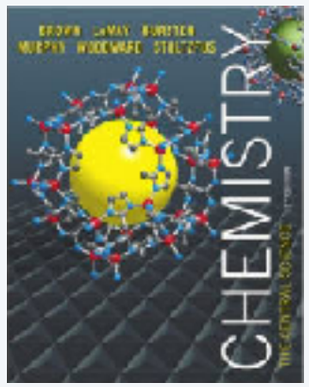
Fundamental challenge for teaching CS (1975-present)

Standard textbooks have been the norm in most fields in the US for decades.

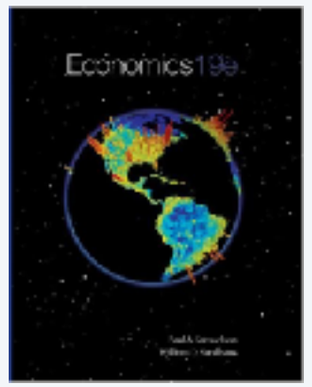
Physics



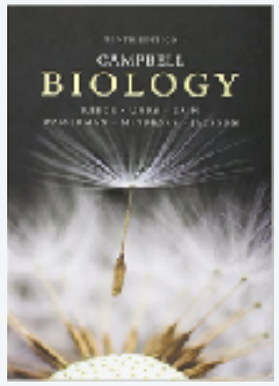
Chemistry



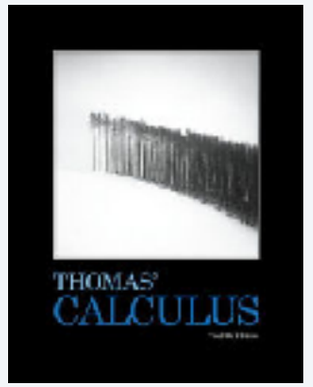
Economics



Biology



Calculus



Computer Science

Central thesis (RS, 1975, 1992)

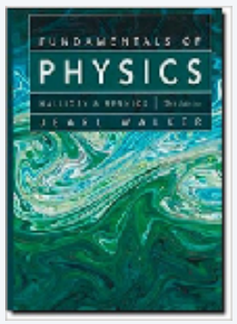
All college students need courses in computer science and algorithms

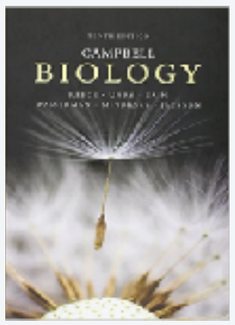
Computer science embraces a significant body of knowledge that is

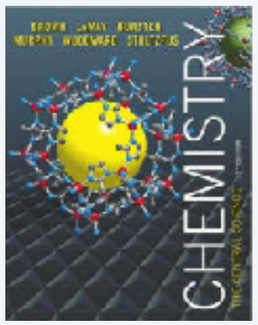
- intellectually challenging
- pervasive in modern life
- critical to modern science and engineering

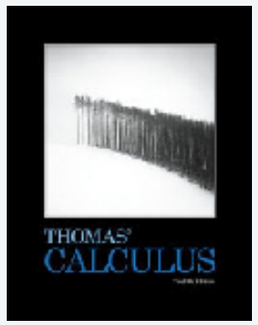
Anyone can learn the importance of

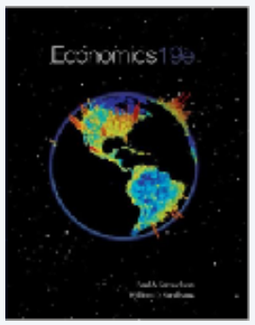
- modern programming models
- the scientific method in understanding program behavior
- algorithms and data structures
- fundamental precepts of computer science
- computation in a broad variety of applications
- preparing for a lifetime of engaging with computation











Goal: A *standard intro text* for CS that can stand alongside other standard intro texts.

[decades of difficult challenges omitted.]

- Identify content
- Change content
- Interface with the computer center
- Choose programming language
- Dot-com bust (enrolls way down)
- Change programming language
- Staffing
- Political battles
- Competing courses
- Inadequate resources
- Abandon computer center
- Financial crash (enrolls way up)
- Windows, OS X, Linux
- …

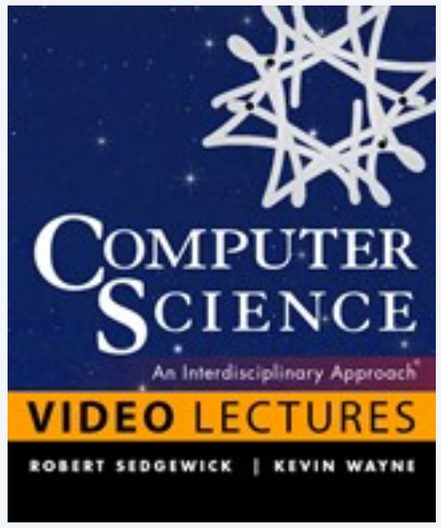
One more course (2015-16)

Computer Science: An Interdisciplinary Approach

- 25 years in development.
- Basis for Princeton's most popular course.

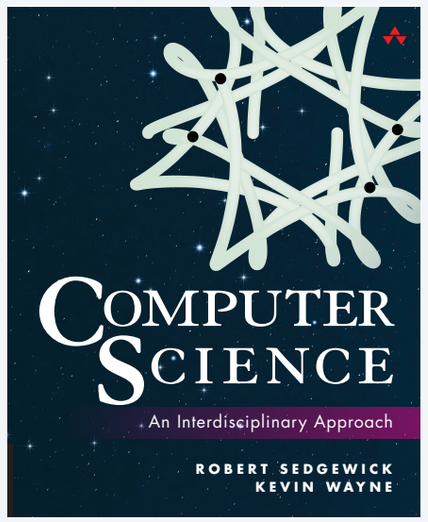
Online lectures (2015)

- 20 lectures, each running 60+ minutes.
- 1000+ state-of-the-art lecture slides.
- Not (yet) a MOOC [long story].
- Available at InformIT (~$30).



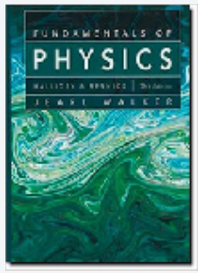
Textbook (2016)

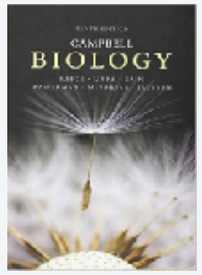
- ~1000 pages
- Programming, algorithms, theory, architecture ...
- Turing, von Neumann, Boole, Shannon ...

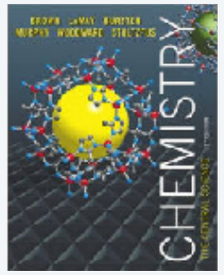


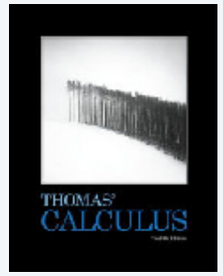
Time to declare victory?

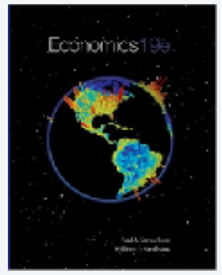
Goal: A *standard intro text* for CS that can stand alongside other standard intro texts.

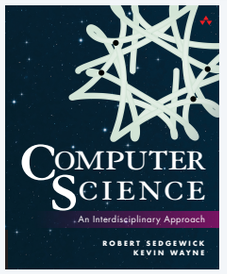






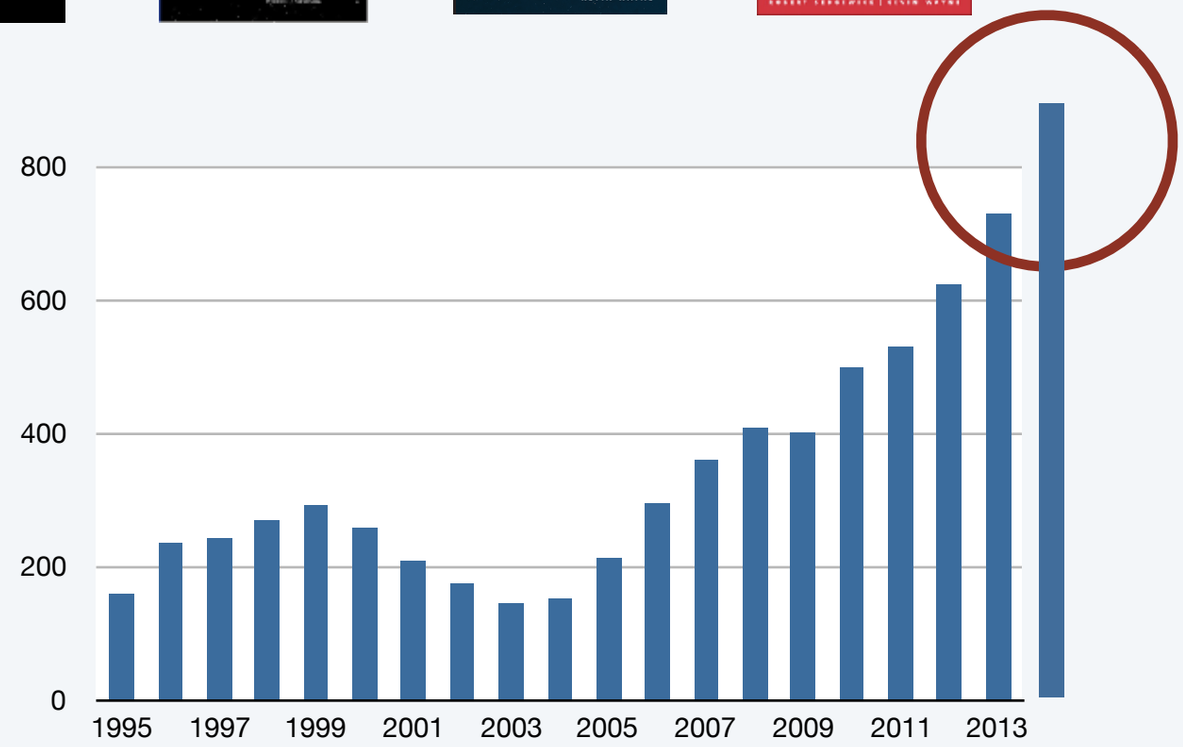


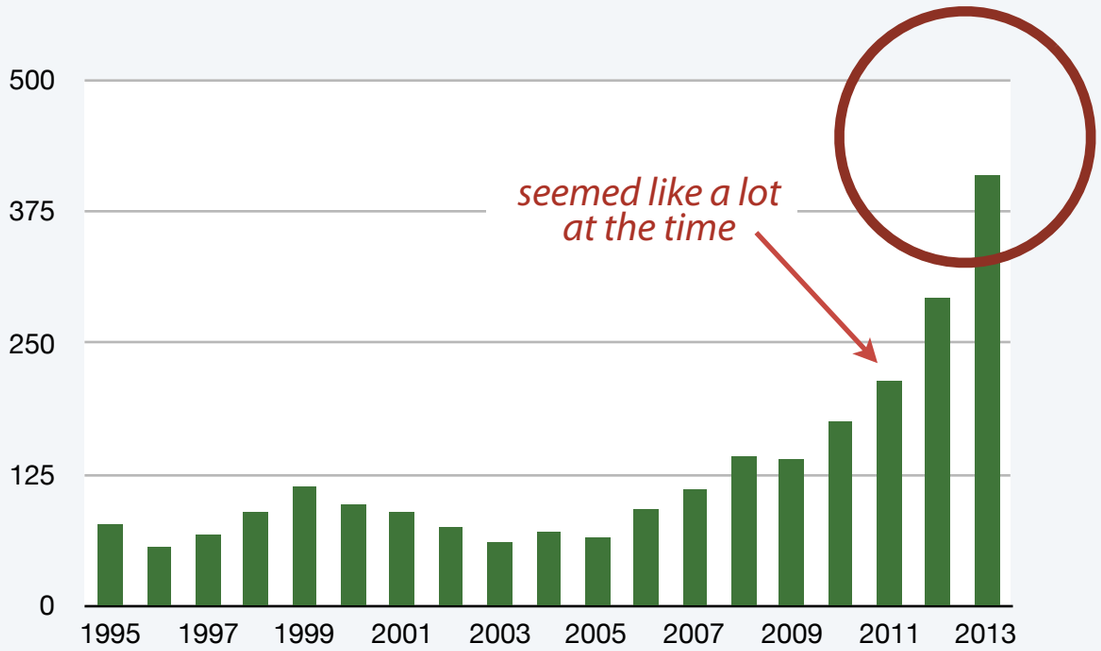




Introduction to CS enrollments

- *Triple* the height of the "bubble"
- 2/3 of all Princeton students
- Largest course at Princeton

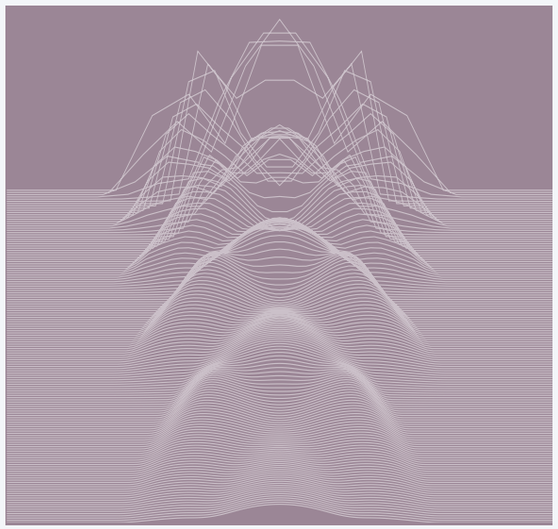




"Algorithms" enrollments

- *Four times* the height of the "bubble".
- 40% of all Princeton students.
- 4th largest course at Princeton

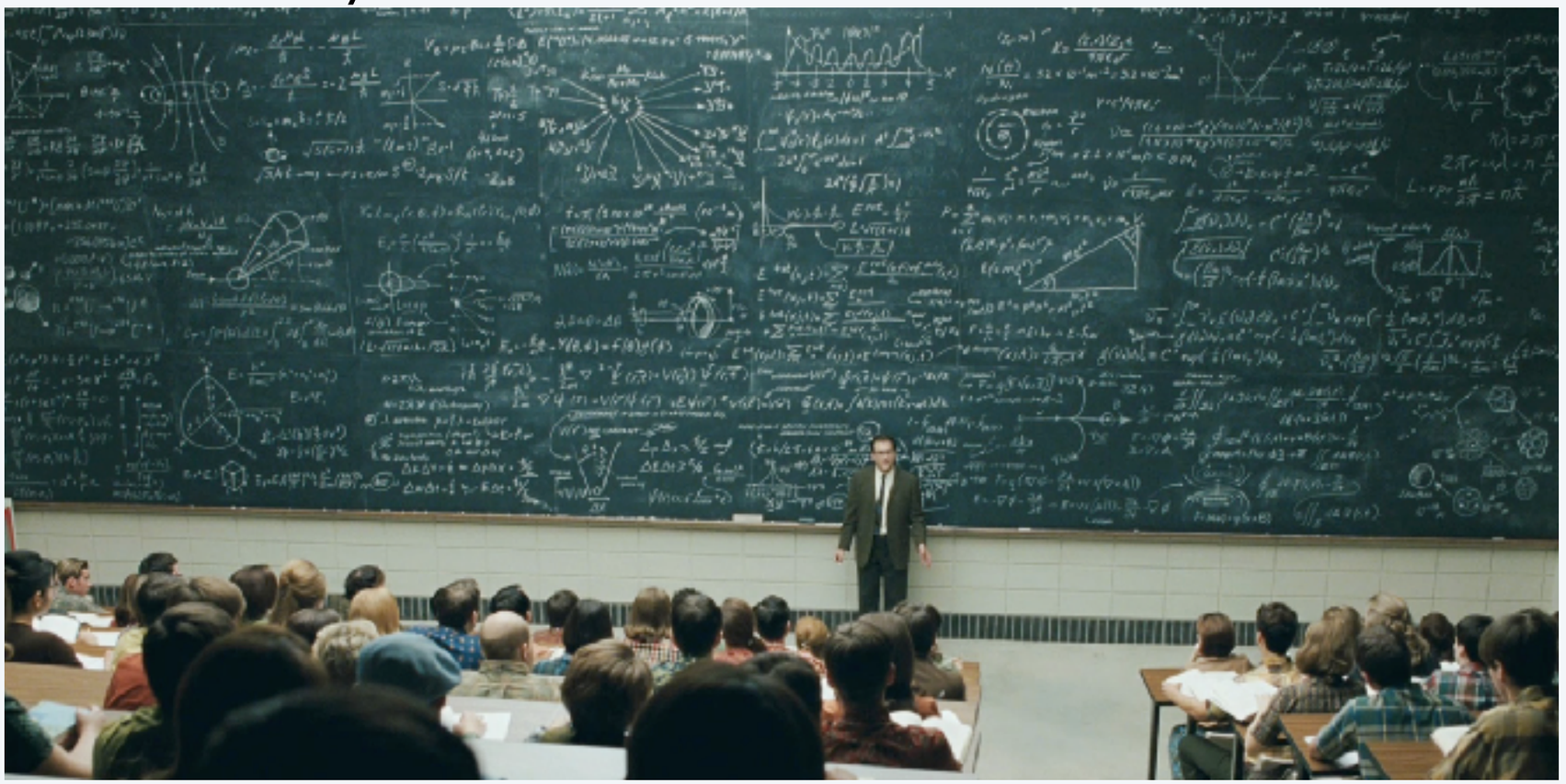
A 21st Century Model for Disseminating Knowledge



- Mission accomplished?
- Taking the plunge
- A way forward
- Postscript

Sudden realization (2015)

20th Century



21st Century

Exactly how are we going to be teaching computer science *at Princeton* in the future?

RS: Hey, we *have* to use the studio-produced lectures!
Everyone else: *Why would we change our biggest and best course?*

details of debate omitted

[Months of difficult negotiations omitted.]

- Students won't watch
- Rules won't permit it
- Will require preparation of new material
- Who will teach it
- How do we change videos?
- Video editing
- Who can watch them?
- Staff will need to reteach
- System won't support it
- Too hard to set up
- …

Last live lecture (September 2015)

Glitches (not unusual)

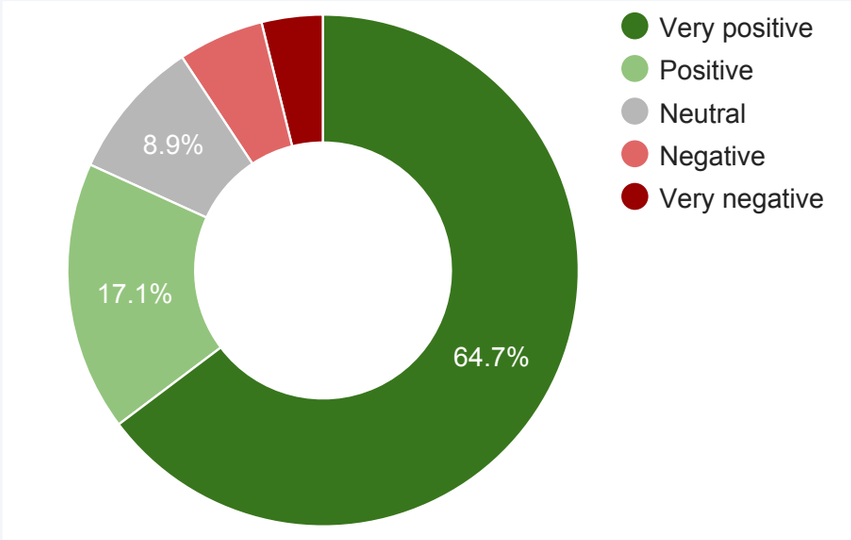
- Over 90 degrees in the room.
- Biggest lecture hall on campus is too small.
- Students in aisles cannot see the screen.
- Sound system stops working halfway through.

Consequence. All students motivated to move online!

An unqualified success

Q. What do you think of the online lectures?

A. 82% of responses were positive+.



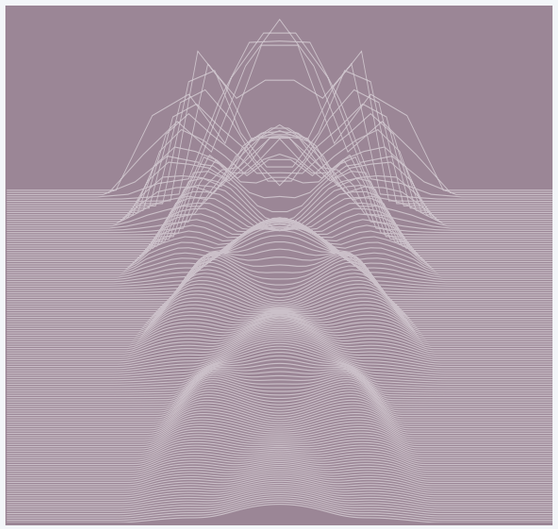
Students *loved* active participation in consuming lecture content

- "Prepares me for a lifetime of active learning online."
- "I like this system, it really lets me go at my own pace and rewatch if I need to."
- "The video lectures are **amazing**. I believe many classes would benefit from this."

Course staff also reaped benefits

- No need to reteach lecture material in office hours.
- More time for interaction with students in small groups.
- More time for interaction in large class meetings.
- Scheduling complications virtually eliminated.

A 21st Century Model for Disseminating Knowledge



- Mission accomplished?
- Taking the plunge
- A way forward
- Postscript

Purpose of the university

is to *produce and disseminate knowledge*

Holy grail for research faculty

- Excellence in teaching.
- Devotion to research.
- *Simultaneously*.

A new model for teaching (this talk)

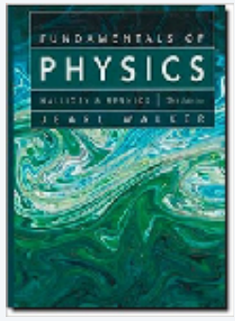
- *Replace live lectures* with online videos.
- *Embrace technology* for efficiency.
- Focus on *helping students succeed*.

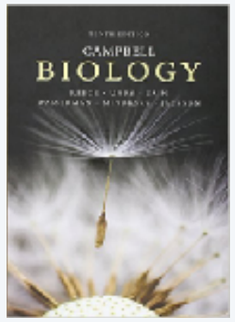
20th-century textbook model

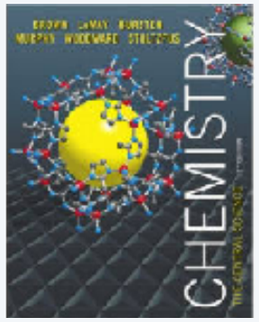
was a standard for introductory courses (in the US) *and is still widely used*

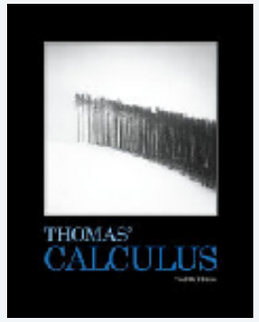
"20th century textbook" model

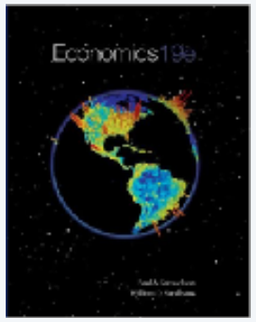
- "Standard" textbooks emerge after significant investment by authors/publishers.
- Distribution model: Teachers "adopt" and students buy textbooks.
- Teachers prepare and deliver lectures (perhaps using author's slides).
- Teachers assess, grade, and certify students.











Pain points

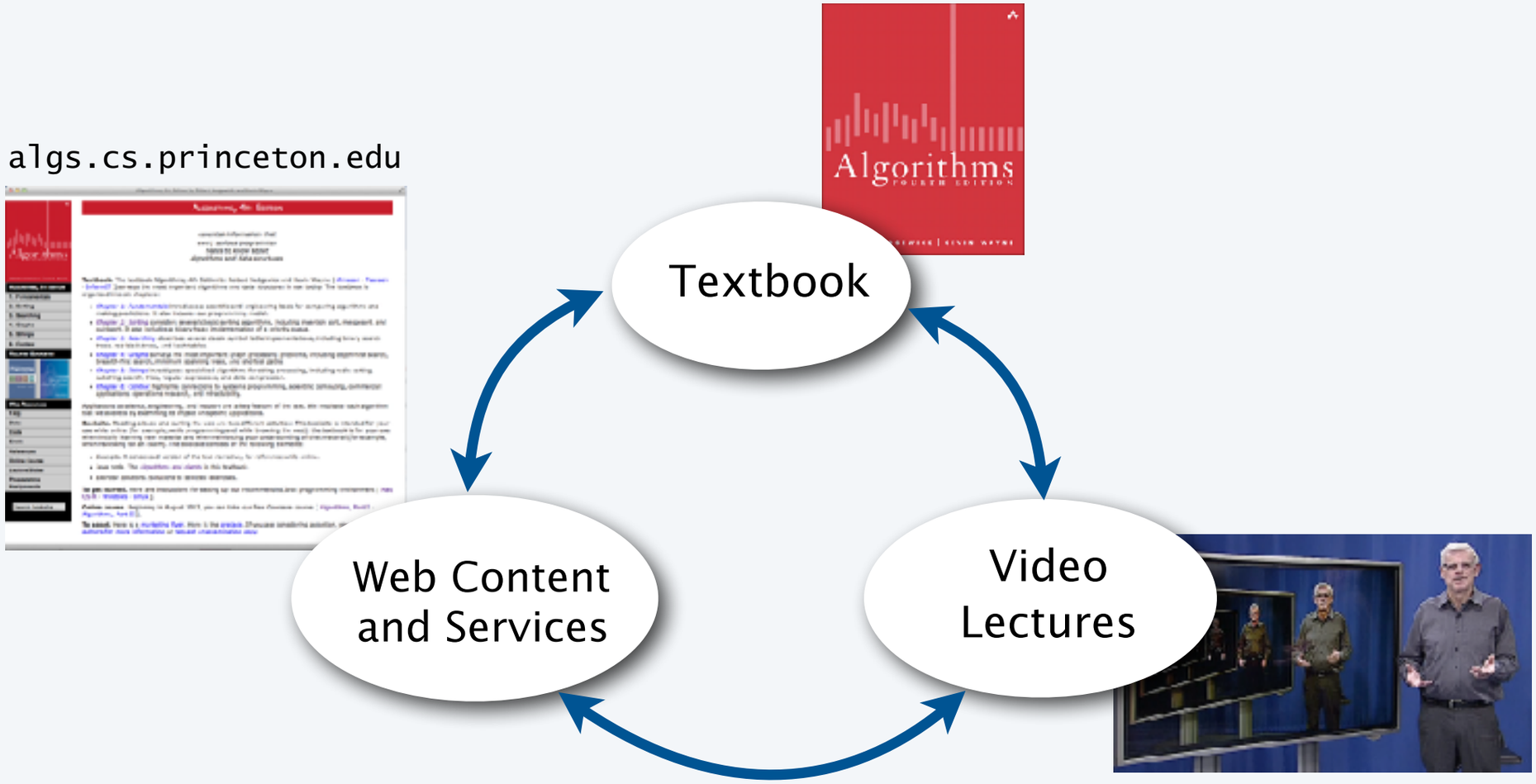
- Inefficiency of adjuncts/professors preparing and delivering "identical" lectures.
- Textbook publishing imploding after move to rental model.
- Passive lecture experience has become unsustainable.
- Assessment efforts generally do not scale.

21st-century textbook model

embraces technology to integrate four abstractions that are *here to stay*

"21st century textbook" model

- Authoritative textbook for use to *learn* and *study* the material.
- Studio-produced video lectures that *introduce* content and *inspire* more study.
- Web content for use to *explore* and *interact* with the material.
- Web services for use by teachers to *assess* and *certify* student learners.



Benefits: Consistent, scalable, and flexible support of *active* teaching/learning.

Algorithms textbook

Algorithms, Fourth Edition

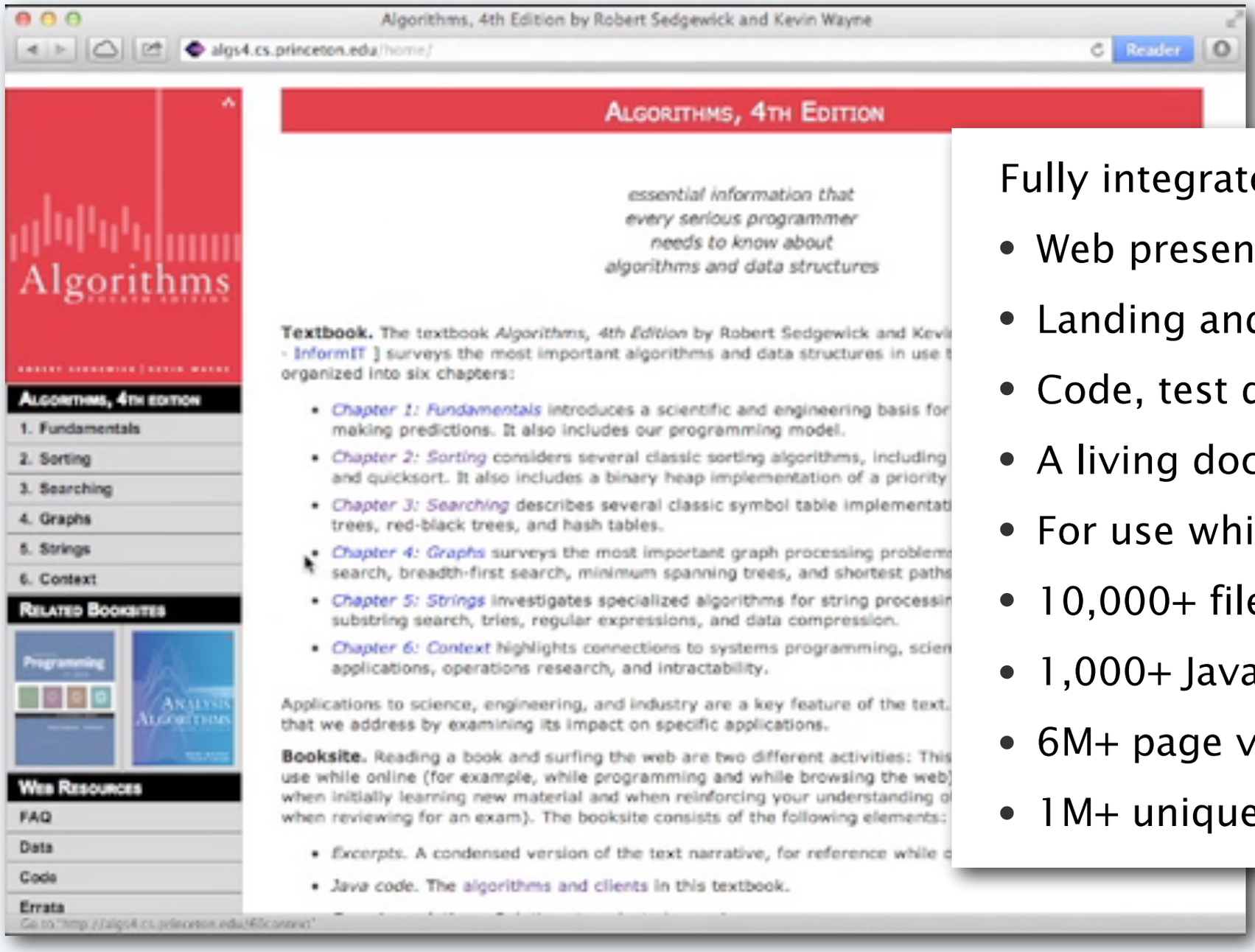
Classic text for decades, 750,000+ sold.

- "Algorithms with code".
- Modern programming model.
- Model course in ACM-IEEE curriculum.
- Completely revamped each decade.
- Widely used around the world.

	edition	*goal for code: clear, readable and*
1970s	1st	*compiles*
1980s	2nd	*runs on examples*
1990s	3rd	*meets performance specs*
2010s	4th	*industrial strength*

Algorithms web content

`http://algs4.cs.princeton.edu/`



Fully integrated with the textbook

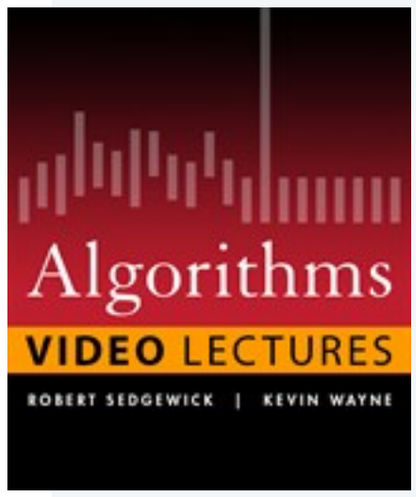
- Web presence.
- Landing and takeoff for search.
- Code, test data, animations.
- A living document.
- For use while coding, exploring.
- 10,000+ files.
- 1,000+ Java programs
- 6M+ page views in the past year
- 1M+ unique users in the past year

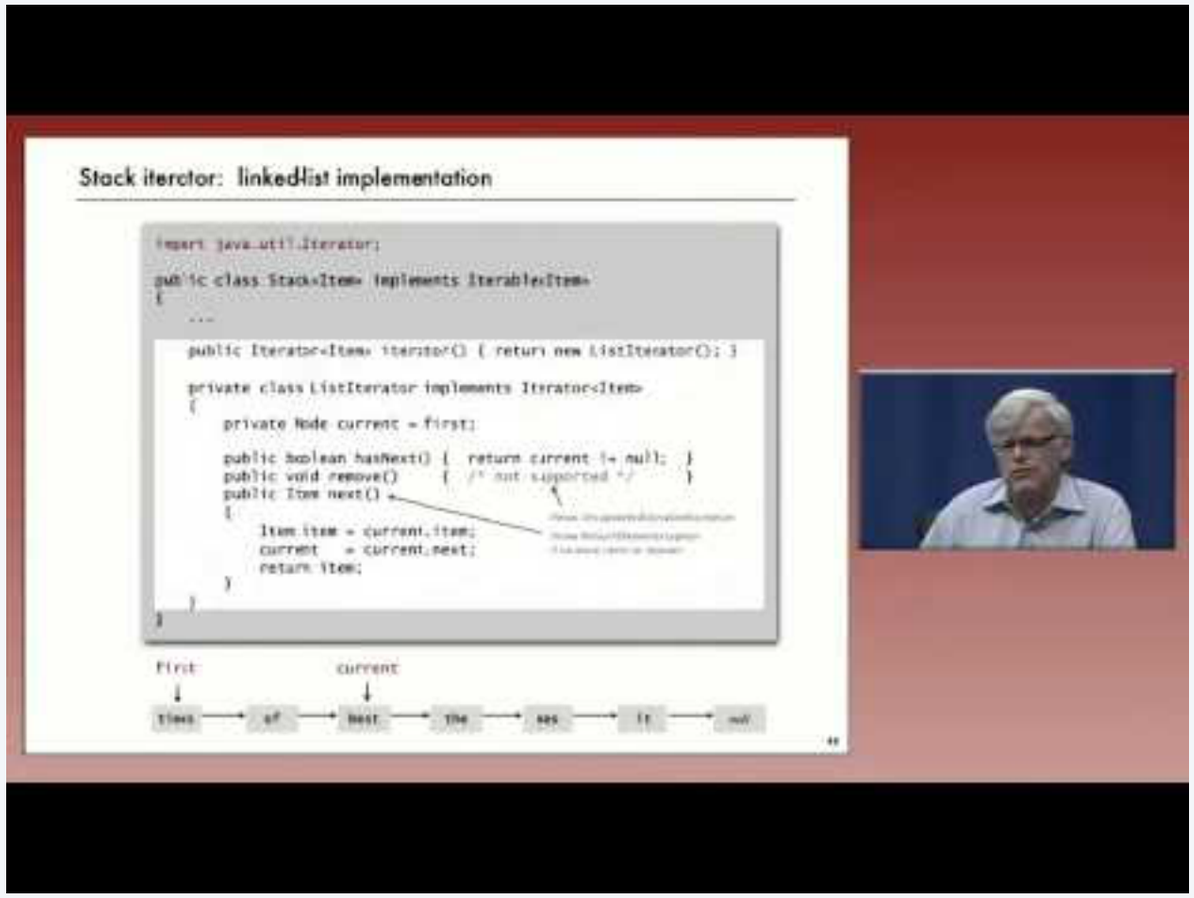
Developed by Kevin Wayne since the mid-2000s

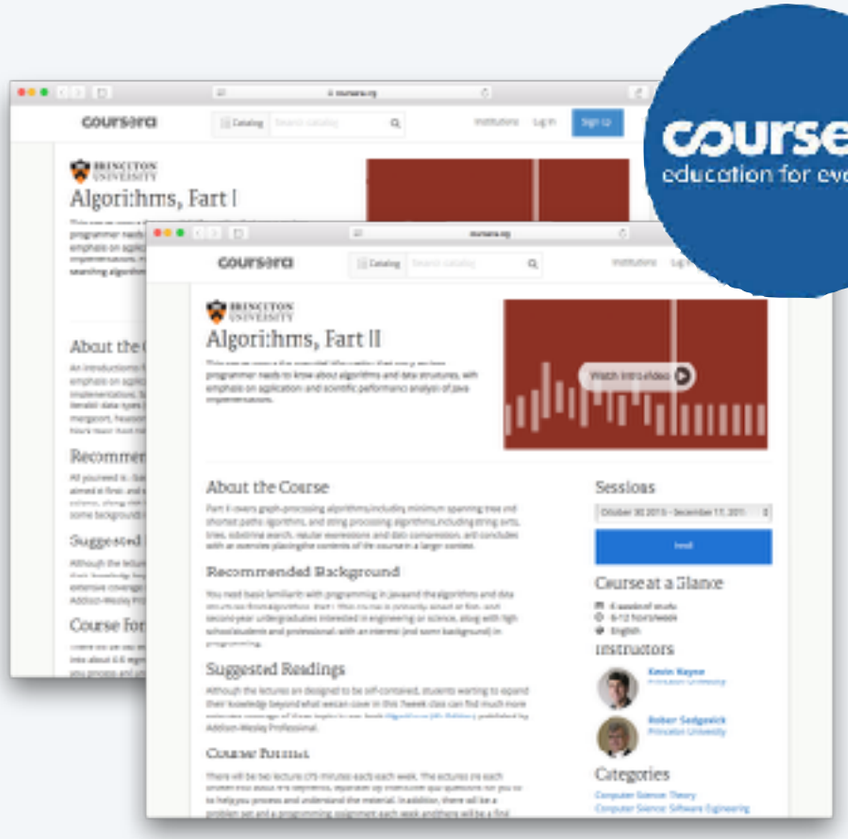
Algorithms online lecture videos

Fully integrated with textbook

- A "top 10 MOOC of all time".
- 24 lectures, about 1.5 hours each.
- Also distributed by Pearson/InformIT.
- Widely used around the world.
- Have reached 1M+ people.







Algorithms web services

Video delivery platform

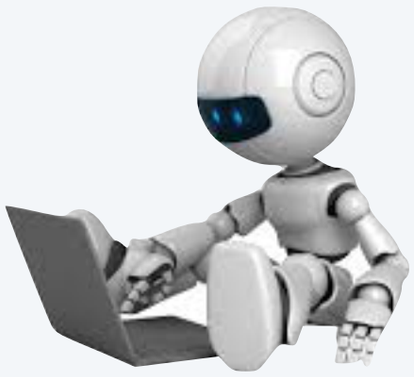
PEARSON informit

Forums and Q&A

piazza
Ask. Answer. Explore. Whenever.

Program assessment infrastructure

- File system/interface for student submissions.
- Dispatch mechanism to support human commentary.
- Used for many CS courses at Princeton.

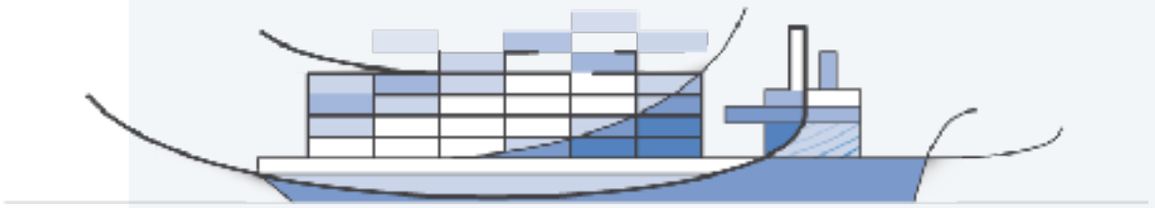


Automated program testing (stay tuned)

- Extensive fine-grained automated testing.
- Correctness (of course).
- Sophisticated performance and probabilistic tests.
- Deployed in an AWS docker container.

Quizzes and exams (stay tuned)

- Random questions drawn from templates.
- Hundreds of templates; millions of questions.
- Auto-graded for self-assessment.
- Web service in a cloud server.



Algorithms automated assessments: quizzes and exams

Example 1: Combinatorial questions

? Every problem in NP is also in P.
F No problem is in both P and NP.
T If P = NP there is a polynomial-time factoring algorithm.
? If P ≠ NP there is a polynomial-time factoring algorithm.
T There is a Turing machine that can decide whether the number of 1s on its input tape is prime.

.
.
.

F The Halting Problem is NP-complete.
T The Traveling Salesperson Problem is NP-complete.
T There is a deterministic Turing machine that can solve every problem in NP.
T There is a DFA that can recognize binary strings that have 1 million 0s and 1 million 1s.
T No polynomial-time algorithm can solve the Halting Problem.

50 facts

8. Computability/Intractability (5 points). For each of the computational problems below, indicate its difficulty by writing the most appropriate choice of *T* (true), *F* (false), or *?* (nobody knows) in the blank at left.

A. _____ Every problem in NP is also in P.

B. _____ There is a DFA that can recognize all binary palindromes.

C. _____ There is a Turing machine that can decide whether the number of 1s on its input tape is prime.

D. _____ No polynomial-time algorithm can solve the Halting Problem.

E. _____ If P = NP there is a polynomial-time factoring algorithm.

$\binom{50}{5}$ = 2M+ questions

Example 2: Data-driven questions

Q17. String sorts. The column on the left is an array of strings to be sorted. The column on the right is in sorted order. The other columns are the contents of the array at some intermediate step during one of the algorithms below. Write the letter corresponding to the correct algorithm under the corresponding column. You will need to use some letters more than once. *Hint* : Do not trace code—think about algorithm invariants.

ghana	spain	aruba	aruba	aruba	chile	ghana	nepal	aruba
sudan	italy	benin	benin	burma	egypt	malta	sudan	benin
wales	ghana	burma	burma	benin	benin	china	qatar	burma
nepal	gabon	china	congo	china	congo	aruba	macau	chile
italy	libya	congo	chile	congo	burma	kenya	aruba	china
malta	macau	chile	china	chile	china	india	yemen	congo
niger	sudan	egypt	egypt	egypt	aruba	libya	niger	egypt
china	india	ghana	gabon	ghana	gabon	burma	wales	gabon
yemen	niger	gabon	ghana	gabon	ghana	zaire	congo	ghana
haiti	chile	haiti	haiti	haiti	haiti	chile	india	haiti
aruba	china	italy	kenya	italy	kenya	haiti	spain	india
kenya	zaire	india	spain	india	spain	nepal	benin	italy
spain	haiti	kenya	india	kenya	india	sudan	chile	kenya
india	wales	libya	libya	libya	libya	yemen	italy	libya
libya	malta	malta	yemen	malta	yemen	spain	burma	macau
burma	yemen	macau	macau	macau	macau	gabon	ghana	malta
macau	congo	nepal	niger	nepal	niger	benin	china	nepal
gabon	benin	niger	malta	niger	malta	congo	gabon	niger
congo	kenya	qatar	italy	qatar	italy	niger	egypt	qatar
benin	nepal	sudan	nepal	sudan	nepal	qatar	zaire	spain
egypt	burma	spain	zaire	spain	zaire	wales	malta	sudan
zaire	qatar	wales	wales	wales	wales	egypt	haiti	wales
chile	aruba	yemen	qatar	yemen	qatar	macau	kenya	yemen
qatar	egypt	zaire	sudan	zaire	sudan	italy	libya	zaire

A								E
	☐	☐	☐	☐	☐	☐	☐	

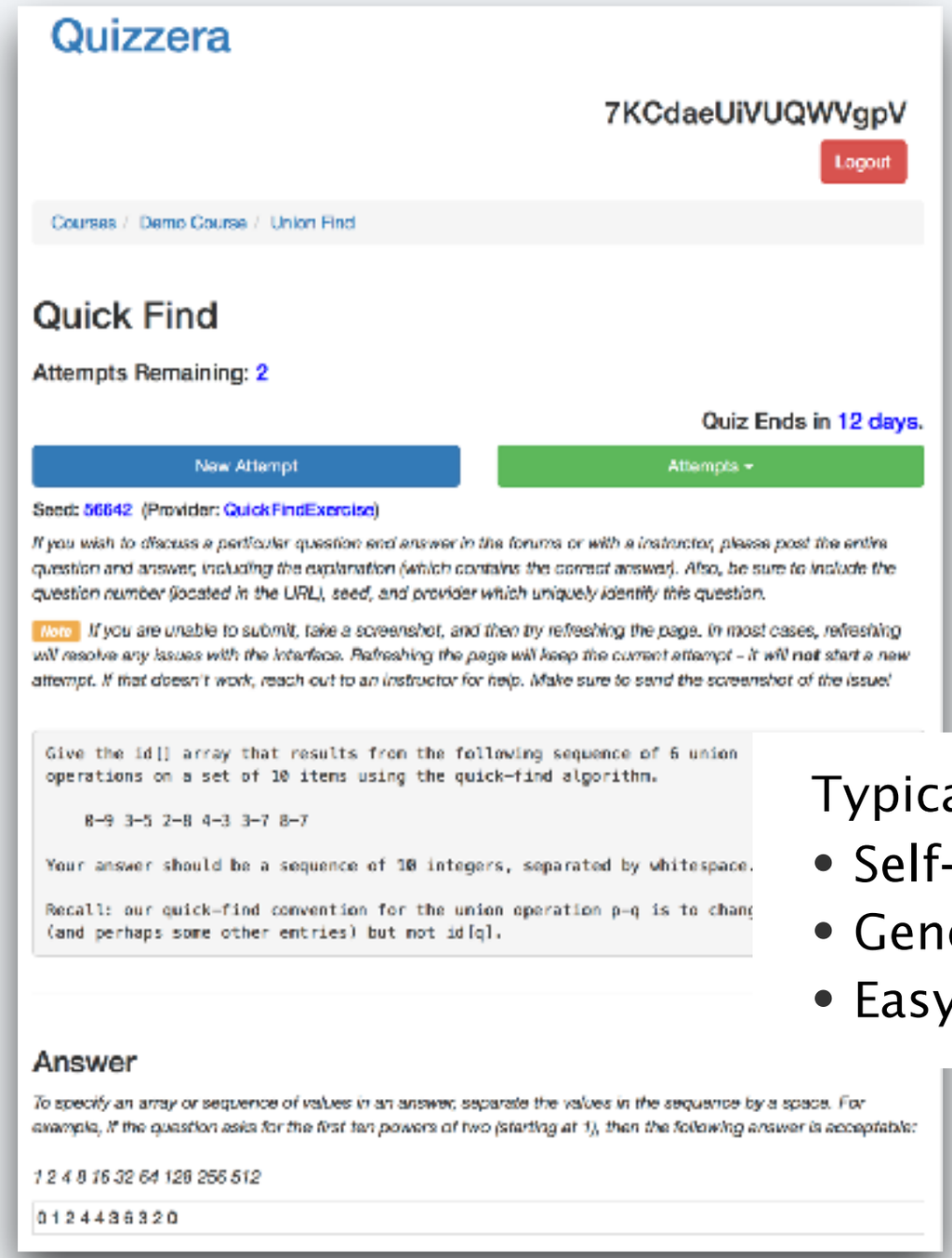
A. *input*
B. *LSD radix sort*
C. *MSD radix sort*
D. *3-way radix quicksort (no shuffle)*
E. *sorted result*

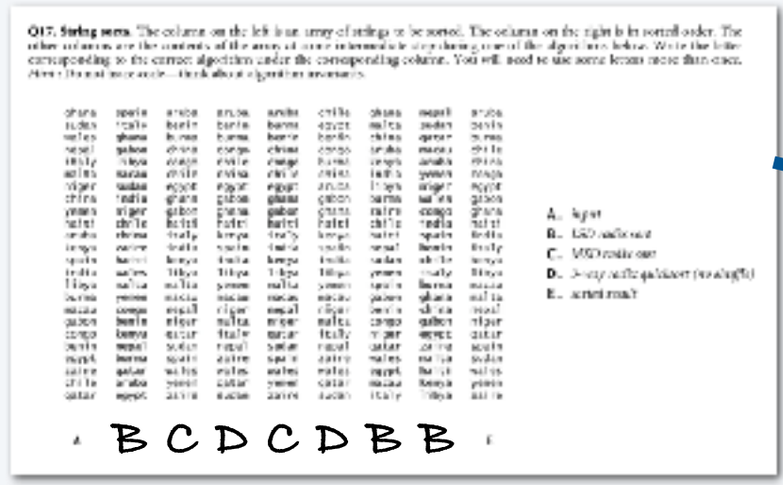
$26^5 \times 24! = 7371780749591669477065359360000$ questions

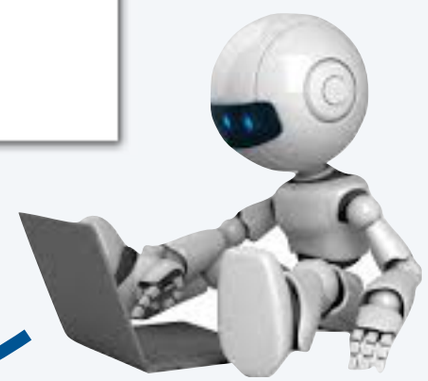
Algorithms automated assessments: quizzes and exams

can be generated and graded in a *fully automatic* fashion.

https://demo.quizzera.io/







7/7

Typical applications

- Self-assessments in a large flipped class.
- Generate huge database of questions for a MOOC.
- Easy to adapt for use in a fully online class.

Quality and consistency of assessments are *dramatically improved* via technology.

Algorithms automated assessments: programs

can be subjected to extensive fine-grained tests and graded *automatically.*

Programs are first checked with best-in-class tools

- Every program must *compile*.
- *Style checks* help develop best-practice programming habits.
- Automatic *bug-finding* is essential ("because it's easy").

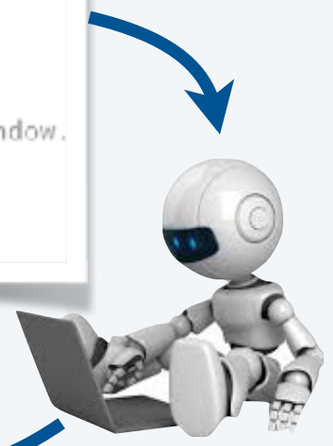
All assignments are based on a fully specified *API*, enabling

- *Correctness checks* (input-output pairs).
- *Timing tests* (essential in an algorithms course).
- *Memory utilization* (also essential in an algorithms course).
- *Probabilistic testing* (for randomized inputs or algorithms.)

Typical applications

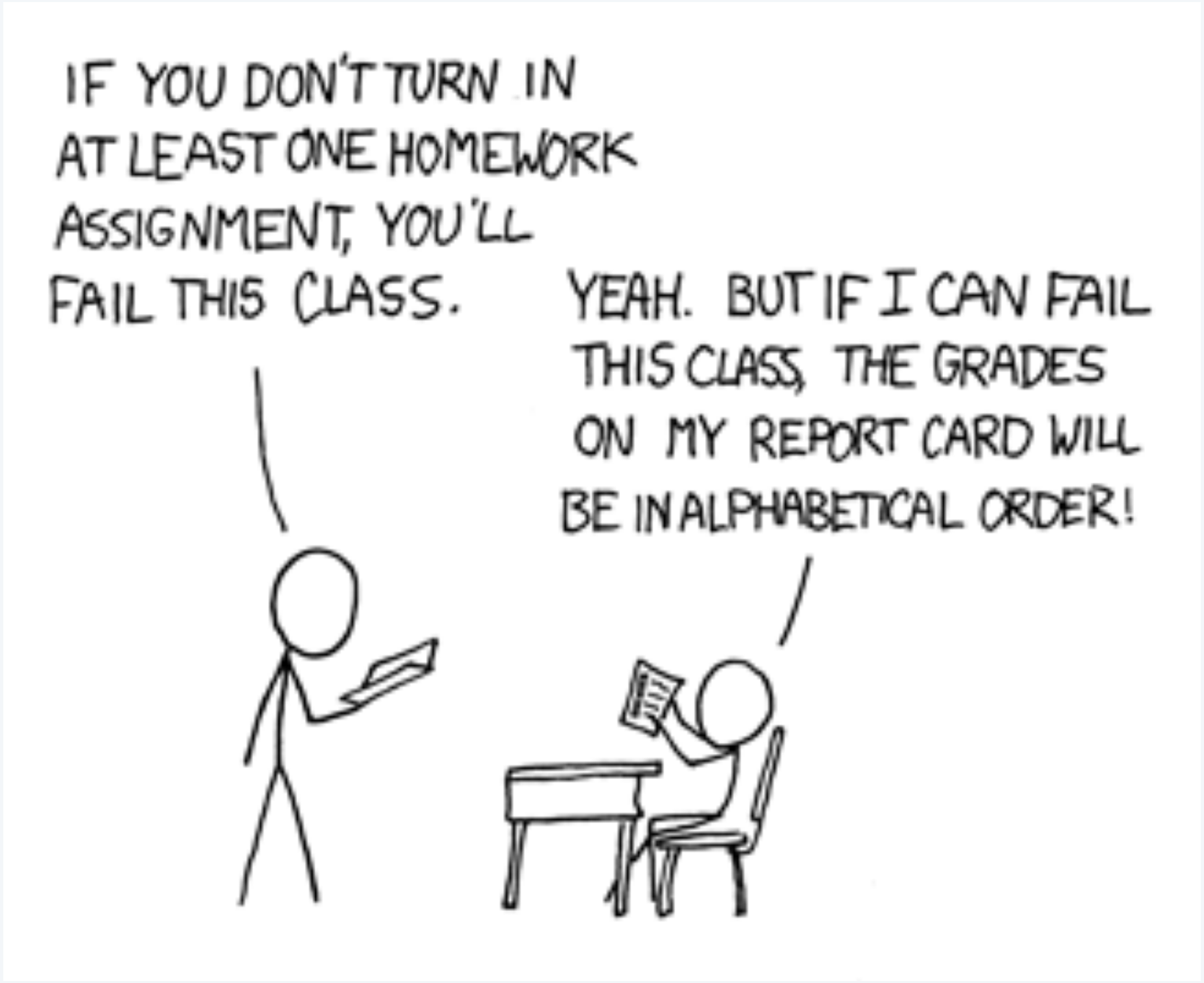
- Grading programs in a large flipped class.
- Grading programs in a MOOC.
- Easy to adapt for use in an online class.

```
public class HelloWorld
{
   public static void main(String[] args)
   {
      // Prints "Hello, World" in the terminal window.
      System.out.println("Hello, World");
   }
}
```



100/100

Quality and consistency of assessments are *dramatically improved* via technology.



https://xkcd.com/336/

Bootstrapping

Courses produce large numbers of qualified students—why not put them to work?

Not-peer grading

- Feedback on code quality is *essential* for beginning programmers.
- Recruit students who have done well in the course to provide it. (they are *not peers*—they have another year of experience coding.
- They can also provide *grades* to supplement automated process.
- Graders expand and reinforce their knowledge by doing so.

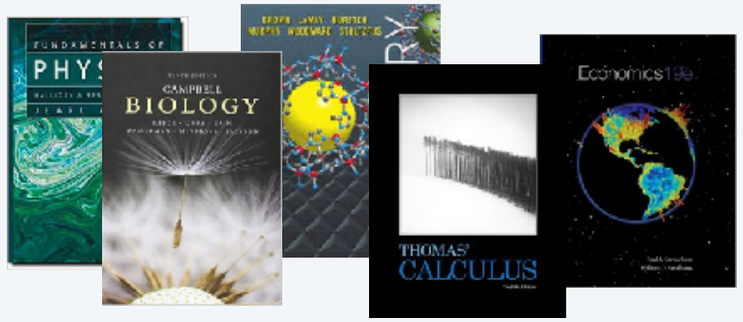
Software development

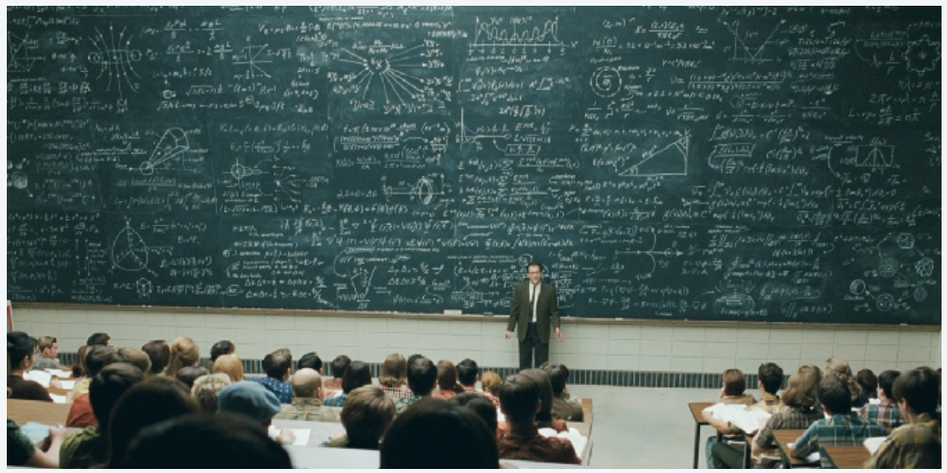
- Best students are *strongly motivated* to create a killer app.
- They also seek independent research projects.
- They also understand the shortcomings of existing software.
- Resulting software tends to be *far better* than otherwise available.

Content development

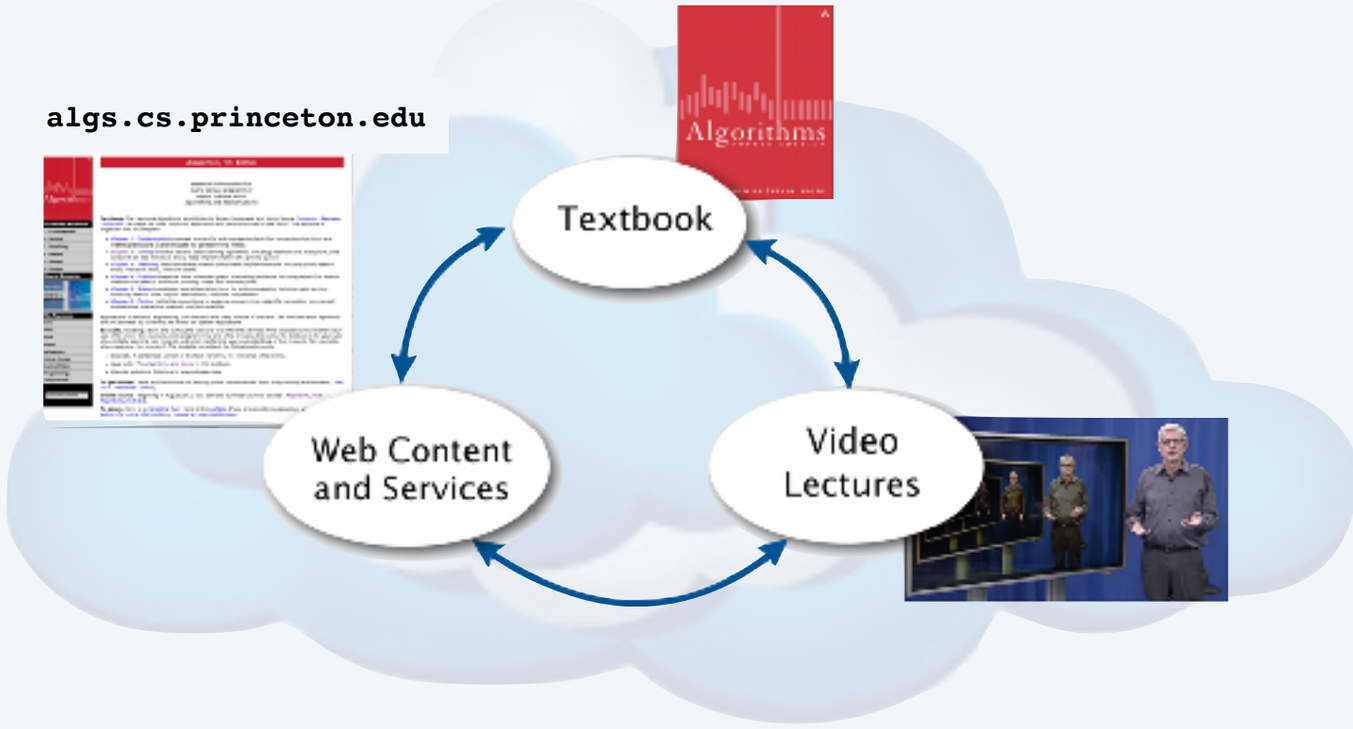
- All students are *strongly motivated* to understand nature of exam questions.
- *Ask them to write questions (and answers) and critique them publicly.*

20th century





21st century

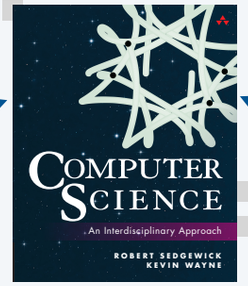


New in 2016: *Computer Science*

Computer Science

- Web content under development since 2000.
  2M+ unique users and 8M+ page views in the past year.
- Studio-produced videos published in 2015.
- Textbook *published June 2016*.

Textbook

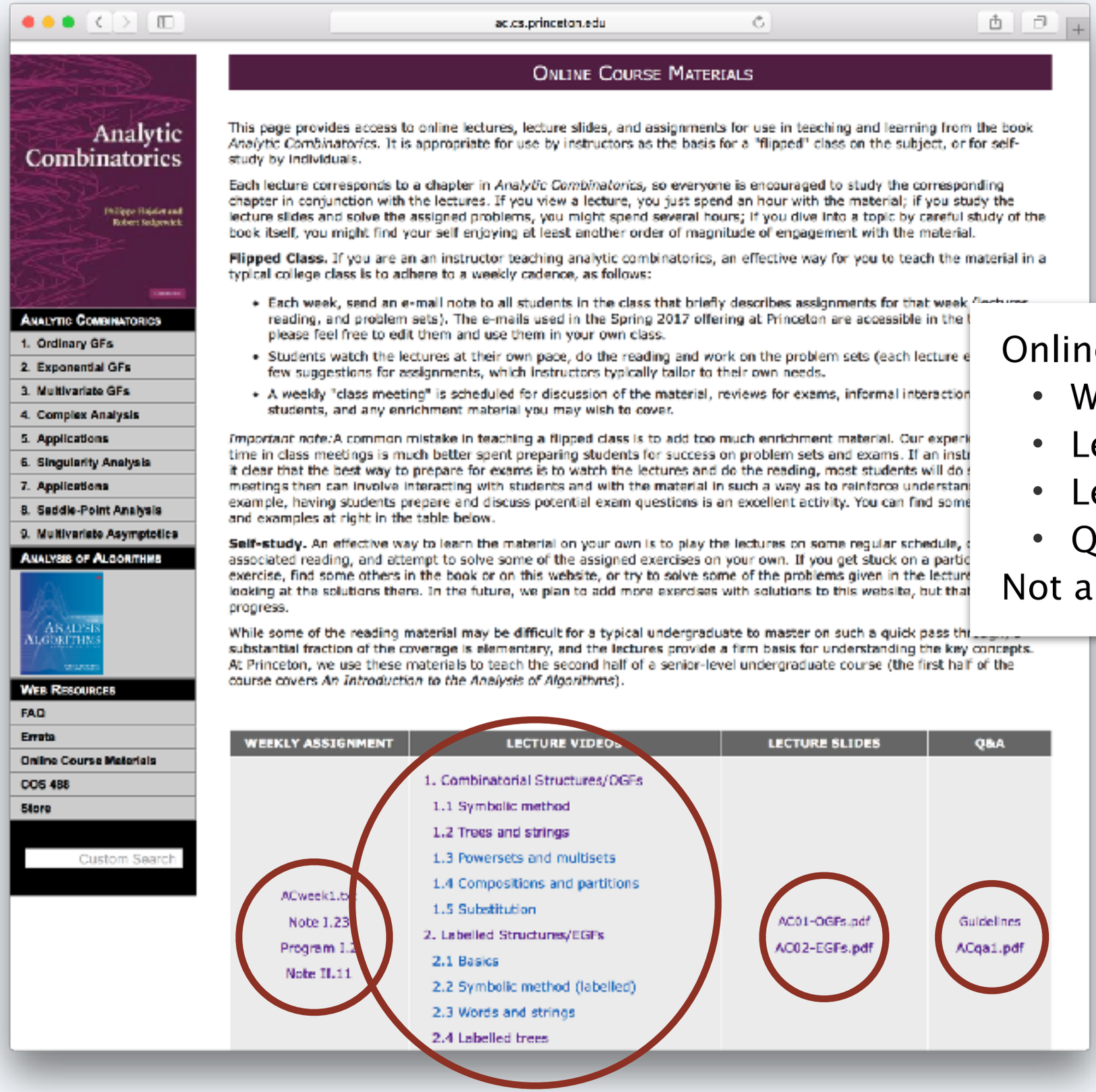


Web content and services

Studio-produced lectures

New in 2017: *Analytic Combinatorics* and *AofA* freely available

`http://ac.cs.princeton.edu/online`



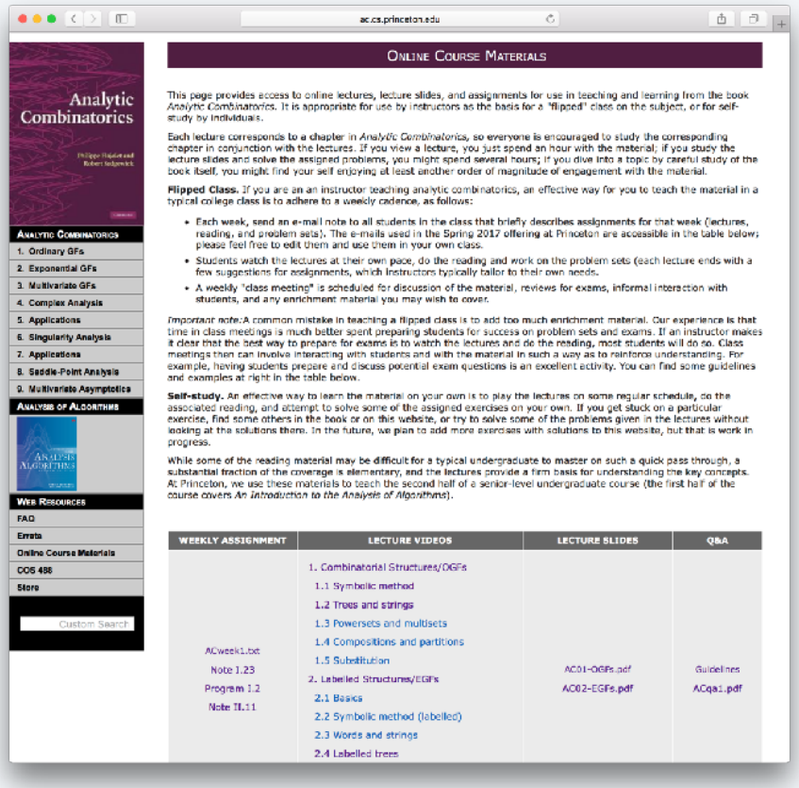
Online course materials

- Weekly assignments
- Lecture videos
- Lecture slides
- Q&A

Not a MOOC

A quick guide to teaching *Analytic Combinatorics*

Visit **`http://ac.cs.princeton.edu/online`** and **`http://aofa.cs.princeton.edu/online`**

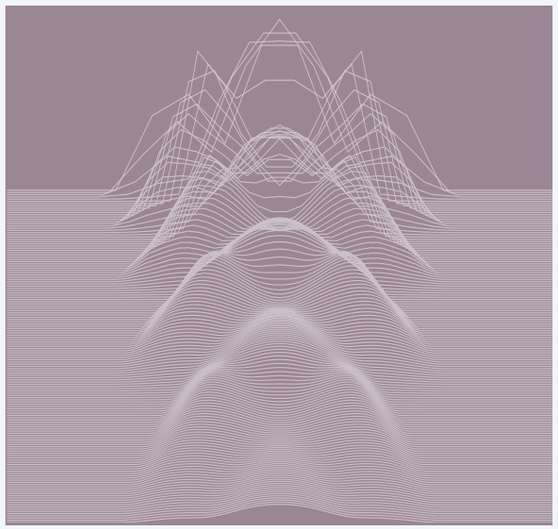


Weekly cadence for professor (2-3 hours per week)

- Send e-mail to students describing lectures and assignments.
- *Students watch lectures online.*
- *Students complete problem sets (as usual).*
- Meet with students to work on prototypical exam questions.
- Monitor online Q&A forum.

Bottom line: Better learning outcomes, more time for research.

A 21st Century Model for Disseminating Knowledge



- Mission accomplished?
- Disruptive changes
- Taking the plunge
- A way forward
- Postscript

A parting thought

(from *John Hennessy* in an interview for an article by Ken Auletta the New Yorker, 2012)

"[Universities,] like newspapers and music companies and much of traditional media a little more than a decade ago are sailing in seemingly placid waters."

"But ... *there's a tsunami coming.*"

What happened to the tsunami?

I think the bloom is now off the rose, and now is going to be the time when some really hard-nosed thinking has to be done about the true value of these online courses.

Shirley Tilghman, 2013

Stumbling blocks

- *Institutions* are trying to take control (and failing).
- Content creation is the province of *individuals*.
- Bad business models, created prematurely.

Result: Plenty of lost opportunities.

RS: Looks like a tsunami to me!

- Tens of thousands of pages of online content
- 100+ hours of lecture videos.
- Reaching millions of individuals.
- 1990s: Lucky to be able to teach my own children.
- 2030s: Will be teaching my own *grandchildren*.

A 21st Century Model for Disseminating Knowledge

Robert Sedgewick
Princeton University

[joint work with Kevin Wayne]