“I Hear and I Forget,

I See and I Remember,

I Do and I Understand.”

- *Chinese Proverb*

*Designing an algorithm is a creative and **rewarding** process*

Think before you leap.

*Designing an algorithm is a creative and **rewarding** process*

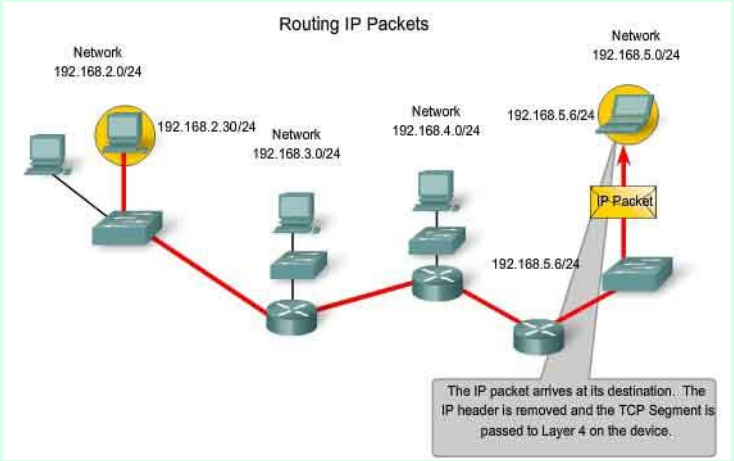
Think before you ~~leap~~ code.

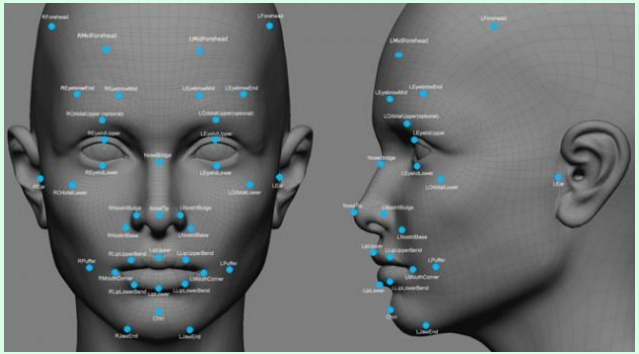
“Weeks of programming can save you hours of planning.”

- *Anonymous quote*

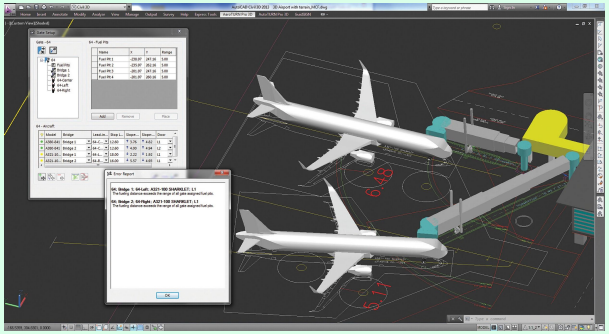
514 × 401 - courses.ischool.utexas.edu

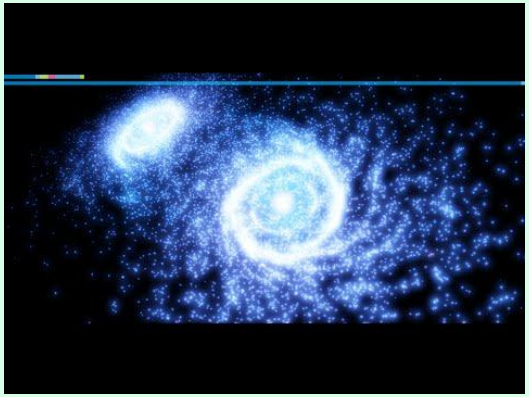
472 × 445 - isoin.es

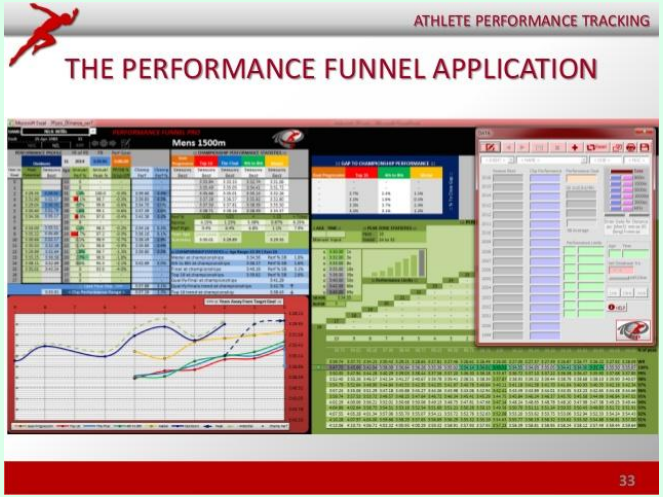


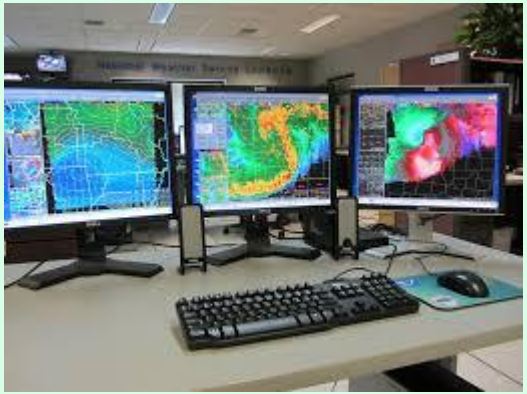


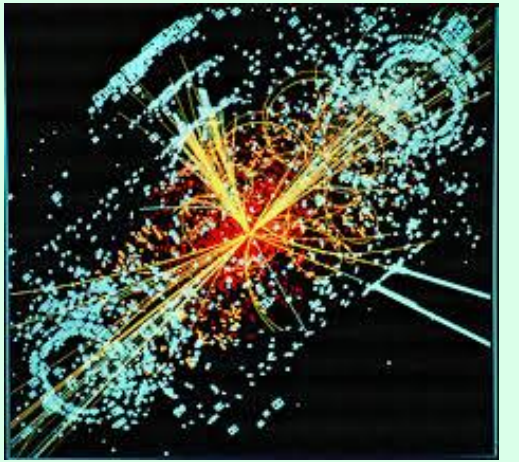
1730 × 1155 - makeawebsitehub.com






ATHLETE PERFORMANCE TRACKING
THE PERFORMANCE FUNNEL APPLICATION
Mens 1500m
33





Why Study Algorithms?

Algorithms make the world run!

Computers are ubiquitous

Education and research
Networks (internet, local nets, ...)
Banks and commerce
Business management
Communication
Defense and military
Health Management
Medical imaging
Exploration (space, sea, land)
Data analytics

Transportation
Multimedia
Robotics
Simulations
Manufacturing
Entertainment
Governance
Security
AI and automation
Social media
...

What are Algorithms?

algorithm

noun

Word used by programmers when they do not want to explain what they did.

What are Algorithms *again*?

“An algorithm is a finite, definite, effective procedure, with some input and some output. ”

- Donald Knuth

An algorithm stays the same whether the program is in java on Windows or C on Linux or python on Mac OS X.

Concept of an algorithm has existed for centuries

Muhammad ibn Musa Al-Khwarizmi formerly latinized as *Algoritmi*

Kitāb al-jabr wa'l-muqābala, which evolved into today's high school algebra text.

“Algorism”: technique of performing arithmetic with Hindu-Arabic numerals developed by al-Khwārizmī.

Famous 9th century Persian mathematician, astronomer, and geographer. (Source: Wikipedia.)

What we'll do in this course

Core computer science problems that arise in many different applications.

- Many problems *reduce* to them.

What we'll do in this course

Follow a design process using a variety of computing problems.

What we'll do in this course

Understand the problem.

What we'll do in this course

Discuss design techniques based on the structure of the problem.

What we'll do in this course

Analyze the algorithm and *discover* efficient solutions to the problem and show it's correct.

How many have **not** taken Data Structures?

Course Syllabus

Lecture Modules

- Algorithm Analysis

Lecture Modules

- Algorithm Analysis
- Graph Algorithms

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - **Greedy Algorithms**

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - Greedy Algorithms
  - **Divide and Conquer**

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - Greedy Algorithms
  - Divide and Conquer
  - **Dynamic Programming**

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - Greedy Algorithms
  - Divide and Conquer
  - Dynamic Programming
  - **Reductions**

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - Greedy Algorithms
  - Divide and Conquer
  - Dynamic Programming
  - Reductions
  - **Network Flow**

Lecture Modules

- Algorithm Analysis
- Graph Algorithms
- Design Techniques
  - Greedy Algorithms
  - Divide and Conquer
  - Dynamic Programming
  - Reductions
  - Network Flow
- NP Complete Problems (if time permits)

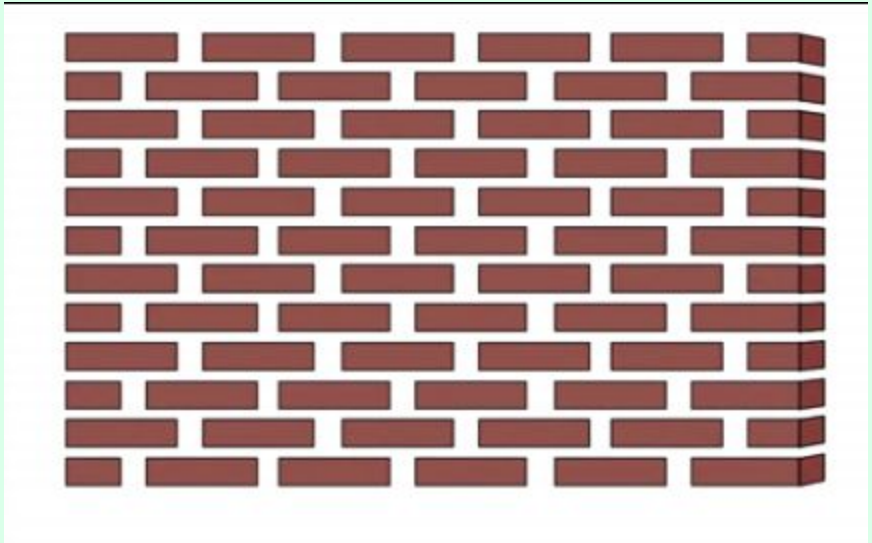
Grading breakup

- Assignments: 25% (No-show for vivas=70% reduction)
- Quizzes, In-class problems, homework: 20%
- Midterm Examination: 25%
- Final Examination: 30%

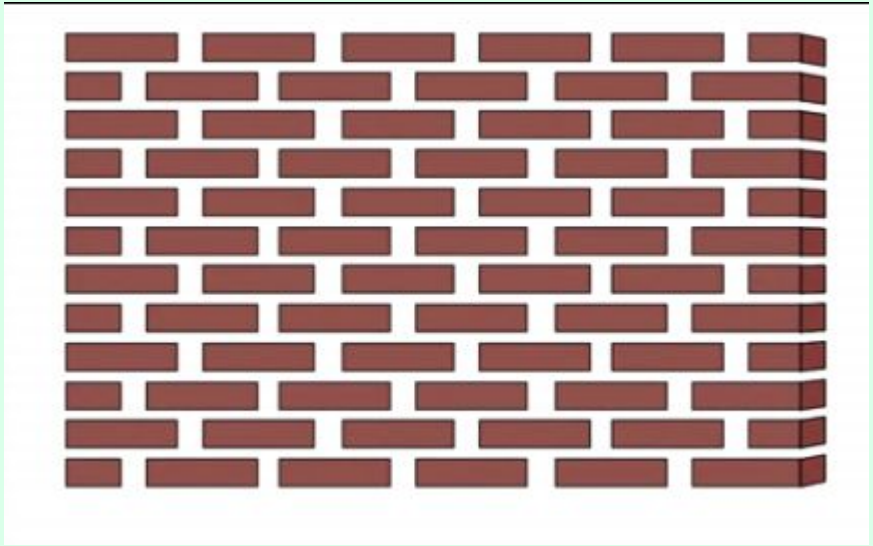
CS 310 TAs (in alphabetical order)

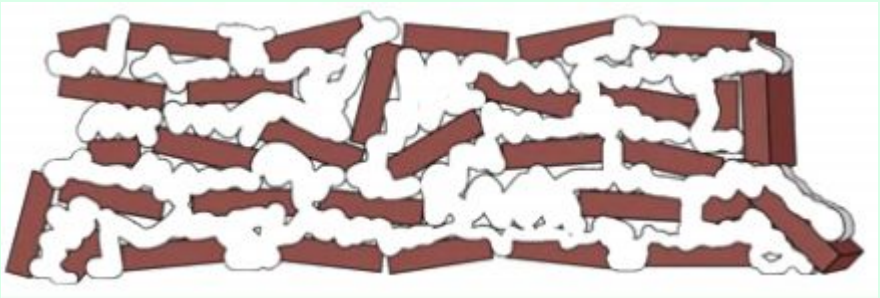
- Ali Ahad (20100284@lums.edu.pk)
- Javaria Hassan (20100058@lums.edu.pk)
- Sheikh Abdul Mannan (20100261@lums.edu.pk)
- Syed Hamza Ahmad (20100108@lums.edu.pk)

*What should you do to **succeed** in this course?*

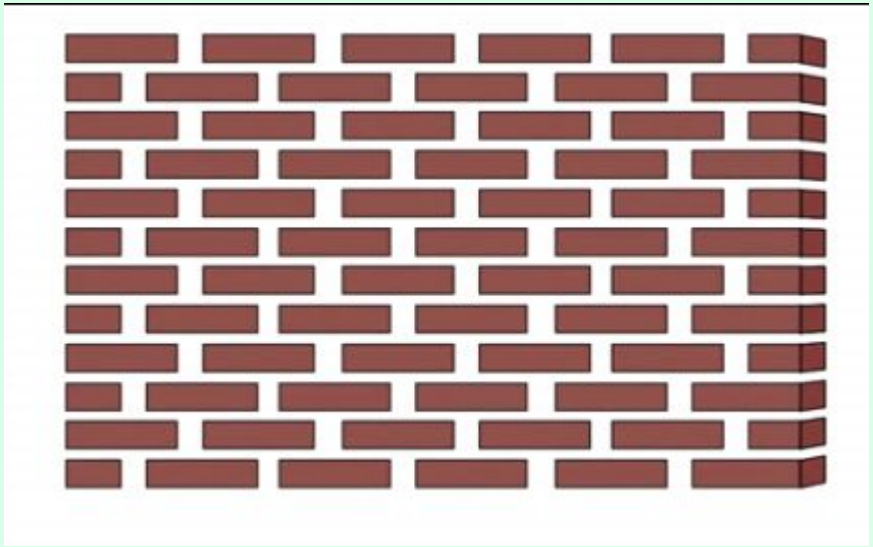


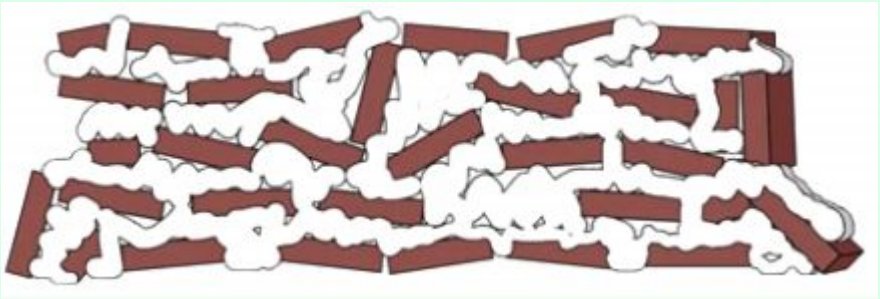
Visual source: Barbara Oakley - Learning how to learn





Visual source: Barbara Oakley - Learning how to learn





Visual source: Barbara Oakley - Learning how to learn

Students who:

- **Enjoy** problem solving and thinking about creative solutions
- **Review material after class** and through spaced repetitions.
- **Focus on understanding** in-class, homework and practice problems
- **Start work on assignments early**

Students who:

- **Cram** for exam a couple of days before it.
- **Don’t review** material after class
- **Waste time** during class
- **Don’t spend sufficient time** on problem solving
- **Procrastinate** on the assignments

*The first time you **actually** understand something is when you can do it yourself.*

*What should you do to **succeed** in this course?*

Come to class on time (*5 minute rule*)

Be an *active* participant.

Enjoy learning!

What things are important in software design?

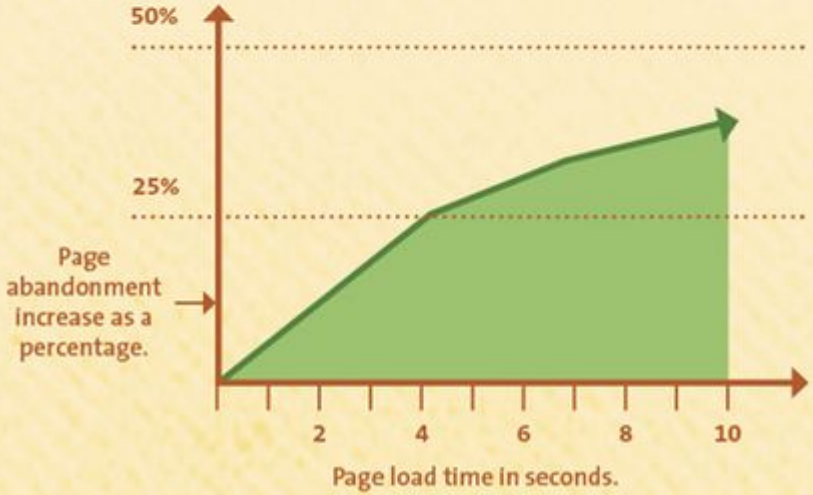
Why is performance (time) important?

How long are you willing to wait for a web page to load?

EVERY SECOND COUNTS

Loading time is a major contributing factor to page abandonment. The average user has no patience for a page that takes too long to load, and justifiably so.

Observation: slower page response time results in an increase in page abandonment, as demonstrated in the following chart.



How do we determine efficiency of an algorithm?

Running Times of different algorithms on inputs of increasing sizes

n $f(n)$	$\lg n$	n	$n \lg n$	n^2	2^n	$n!$
10	0.003 μs	0.01 μs	0.033 μs	0.1 μs	1 μs	3.63 ms
20	0.004 μs	0.02 μs	0.086 μs	0.4 μs	1 ms	77.1 years
30	0.005 μs	0.03 μs	0.147 μs	0.9 μs	1 sec	8.4×10^{15} yrs
40	0.005 μs	0.04 μs	0.213 μs	1.6 μs	18.3 min	
50	0.006 μs	0.05 μs	0.282 μs	2.5 μs	13 days	
100	0.007 μs	0.1 μs	0.644 μs	10 μs	4×10^{13} yrs	
1,000	0.010 μs	1.00 μs	9.966 μs	1 ms		
10,000	0.013 μs	10 μs	130 μs	100 ms		
100,000	0.017 μs	0.10 ms	1.67 ms	10 sec		
1,000,000	0.020 μs	1 ms	19.93 ms	16.7 min		
10,000,000	0.023 μs	0.01 sec	0.23 sec	1.16 days		
100,000,000	0.027 μs	0.10 sec	2.66 sec	115.7 days		
1,000,000,000	0.030 μs	1 sec	29.90 sec	31.7 years		