Randomness in Computing

CS 237

LECTURE 1

Randomness in Computing

- Course information
- Uses of probability in CS
- Verifying polynomial identities

Sofya Raskhodnikova and Wayne Snyder

Course information

1. Course staff
2. Course website(s)
3. Piazza bonus
4. Prerequisites
5. Textbook(s)
6. TopHat
7. Syllabus
8. Homework logistics
9. Collaboration policy
10. Exams and grading

CS 237

Tips for the course

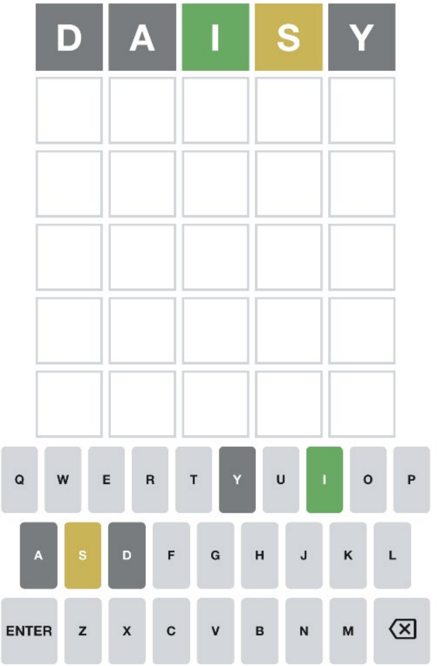
- Concepts in this course take some time to sink in: be careful not to fall behind.
- Prepare for each lecture by reviewing material from the previous lecture and doing assigned reading.

- Attend the *lectures*: some material will be presented on the blackboard (and some of it is not in the book).
- Attend the *discussions*: practice problem solving.
- Take advantage of *office hours*.

- Be active in lectures, discussions, and on piazza.
- Study with a friend: do exercises and quiz each other.
- Allocate lots of time for the course: comparable to a project course, but spread more evenly.

Tips for the course: HW

- Start working on HW early.
- Spread your HW time over multiple days.
- You can work in groups (up to 4 people), but spend 1-2 hours thinking about it on your own before your group meeting.



Tips: learning problem solving

To learn problem solving, you have to do it:

- Try to think how you would solve any presented problem before you read/hear the answer.
- Do exercises in addition to HW
  - do solved exercises in the supplementary textbook

CS 237

Tips: how to read a math text

- Not like reading a mystery novel.
- The goal is not to get the answers, but to learn the techniques.
- Always try to foresee what is coming next.
- Always think how you would approach a problem before reading the solution.
- This applies to things that are not explicitly labeled as problems.

Picture source: pngtree.com

CS 237

Skills we will work on

- Mathematical reasoning
- Expressing your ideas
  - abstractly (suppress inessential details)
  - precisely (rigorously)
- Probabilistic thinking
- A bit of algorithmic thinking
- Problem solving
- Computer simulations of probabilistic experiments
- Having **FUN** with all of the above!!!

Could they ask me questions

about CS 237 material on job interviews?

- You bet.

Uses of Probability in Computing

- To speed up algorithms.
- To enable new applications:
  - Symmetry breaking in distributed algorithms, cryptography, privacy, online games and gambling.
- To simulate real world events in physical systems: model them as happening randomly.
- To analyze algorithms when data is generated from some distribution:
  - learning theory, data compression.
- To analyze algorithms when errors happen randomly
  - error-correcting codes.
- Analyzing statistics from sampling.

Probability in CS Curriculum

- CS 350: Fundamentals of Computing Systems
- CS 507: Introduction to Optimization in Computing and Machine Learning
- CS 542: Machine Learning
- CS 535: Complexity Theory
- CS 537: Randomness in Computing
- CS 558: Introduction to Network Security
- …

Verifying Polynomial Identities

- $(x+1)(x-2)(x+3)(x-4)(x+5)(x-6) \stackrel{?}{\equiv} x^6 - 7x + 37$

Task: Given two polynomials $f(x)$ and $g(x)$, verify if $f(x) \equiv g(x)$.

Polynomials

A polynomial in variable x is a function of the form

$$p(x) = a_d x^d + a_{d-1} x^{d-1} + \cdots + a_0.$$

degree $d \geq 0$ and integer

variable

coefficients

Polynomial	Degree	Example	General form
Constant	0	3	$p(x) = \boldsymbol{a}$
Linear	1	$-7x - 2$	$p(x) = \boldsymbol{a}x + \boldsymbol{b}$
Quadratic	2	$x^2 - 4x + 3$	$p(x) = \boldsymbol{a}x^2 + \boldsymbol{b}x + \boldsymbol{c}$
Cubic	3	$x^3 - 1$	$p(x) = \boldsymbol{a}x^3 + \boldsymbol{b}x^2 + \boldsymbol{c}x + \boldsymbol{d}$

Roots of a Polynomial

A polynomial in variable x is a function of the form

degree

$$p(x) = a_d x^{d} + a_{d-1} x^{d-1} + \cdots + a_0 \, .$$

- Number r is a **root** of $p(x)$ if $p(r) = 0$.

  Ex. $p(x) = x^2 - 9$ has two roots

  namely, 3 and -3.

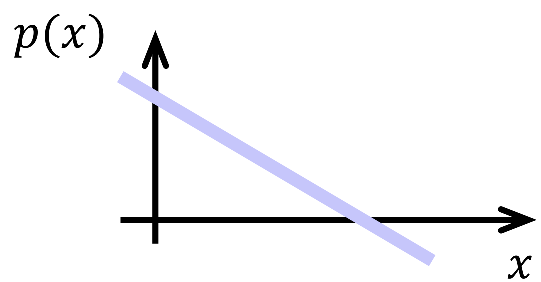
Roots of a Polynomial

A polynomial in variable x is a function of the form

degree

$$p(x) = a_d x^d + a_{d-1} x^{d-1} + \cdots + a_0 \,.$$

- Number r is a **root** of $p(x)$ if $p(r) = 0$.
- A linear function has at most 1 root.



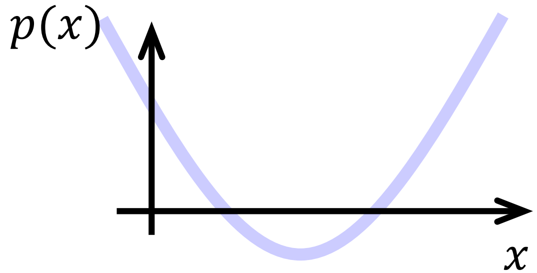
CS 237 Roots of a Polynomial

A polynomial in variable x is a function of the form

degree

$$p(x) = a_d x^{d} + a_{d-1}x^{d-1} + \cdots + a_0\,.$$

- Number r is a **root** of $p(x)$ if $p(r) = 0$.
- A quadratic function has at most 2 roots.



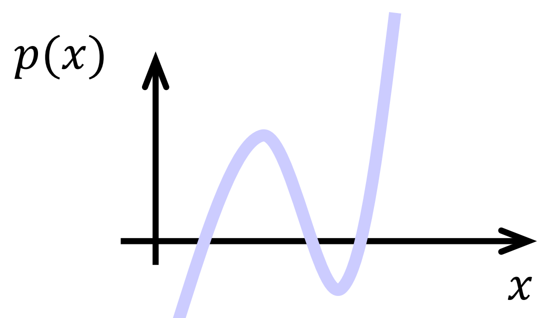
Roots of a Polynomial

A polynomial in variable x is a function of the form

degree

$$p(x) = a_d x^d + a_{d-1} x^{d-1} + \cdots + a_0 \,.$$

- Number r is a **root** of $p(x)$ if $p(r) = 0$.
- A cubic function has at most 3 roots



Roots of a Polynomial

A polynomial in variable x is a function of the form

degree

$$p(x) = a_d x^{d} + a_{d-1} x^{d-1} + \cdots + a_0 \,.$$

- Number r is a **root** of $p(x)$ if $p(r) = 0$.

Fundamental Theorem of Algebra

A polynomial of degree d has at most d roots.

Verifying Polynomial Identities

- $(x+1)(x-2)(x+3)(x-4)(x+5)(x-6) \stackrel{?}{\equiv} x^6 - 7x + 37$

Task: Given two polynomials $f(x)$ and $g(x)$, verify if $f(x) \equiv g(x)$.

no use of randomness

Idea 1 (deterministic): Convert both polynomials to canonical form

$$a_d x^d + a_{d-1} x^{d-1} + \cdots + a_0.$$

- **It is slow:** If $f(x)$ is given as $(b_1 x - c_1) \cdot \ldots \cdot (b_d x - c_d)$ conversion by consecutively multiplying monomials requires about d^2 multiplications of coefficients.

(Faster with Fourier Transform)

End of lecture 1

- Next time, we will see a method for verifying polynomial identities using randomness.

Verifying Polynomial Identities

Task: Given two polynomials $f(x)$ and $g(x)$, verify if $f(x) \equiv g(x)$.

Observation: Let $p(x) = f(x) - g(x)$.

Then we need to verify if $p(x) \equiv 0$.

Idea 2 (randomized): Evaluate $p(x)$ on random integers.

All numbers in this set are equally likely

Or, d could be any upper bound on the degree of $p(x)$, that is, $d \geq$ degree $p(x)$.

Let d = max degree of $f(x)$ and $g(x)$

1. Pick r uniformly from $\{1, \ldots, 100d\}$.
2. Compute $p(r) = f(r) - g(r)$
3. **reject** if $p(r) = 0$; o. w. **accept**.

Only $d - 1$ multiplications needed to evaluate a product of d monomials

- Does this procedure accept or reject for our example when $r = 2$?

$$(x+1)(x-2)(x+3)(x-4)(x+5)(x-6) \stackrel{?}{\equiv} x^6 - 7x + 37$$

Analysis of Correctness

Task: Given two polynomials $f(x)$ and $g(x)$, verify if $f(x) \equiv g(x)$.

Let d = max degree of $f(x)$ and $g(x)$

1. Pick r uniformly from $\{1, \dots, 100d\}$.
2. Compute $p(r) = f(r) - g(r)$
3. **reject** if $p(r) = 0$; o. w. **accept**.

- If $f(x) \equiv g(x)$, we always accept.

No error in this case

- Otherwise, consider (non-zero) polynomial $p(x)$.
  - It has degree at most d.
  - By Fundamental Theorem of Algebra $p(x)$ has at most d roots.
  - We accept (incorrectly) only if we picked r to be a root of $p(x)$
  - This happens in at most d out of $100d$ cases, i.e., with probability at most

$$\frac{d}{100d} = \frac{1}{100}$$

.01 probability of error

How can we make it even smaller?