

COL 351 : ANALYSIS & DESIGN OF ALGORITHMS

LECTURE 1

INTRODUCTION TO ALGORITHMS

JULY 24, 2026

|

ROHIT VAISH

WHAT IS AN ALGORITHM?

WHAT IS AN ALGORITHM ?



WHAT IS AN ALGORITHM ?



Any piece of code ?

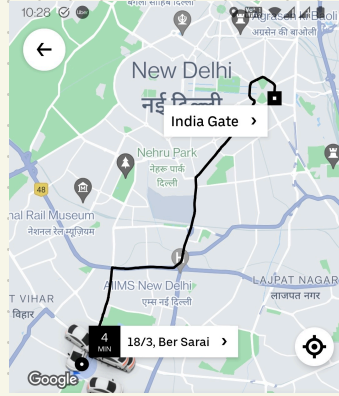
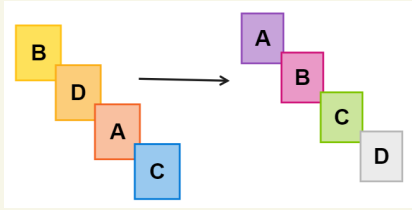
WHAT IS AN ALGORITHM?



Any piece of code?

♥ A recipe for solving some computational problem.

EXAMPLES



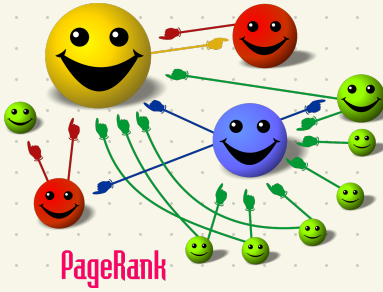
Sorting a list
of objects

finding shortest path
between two places

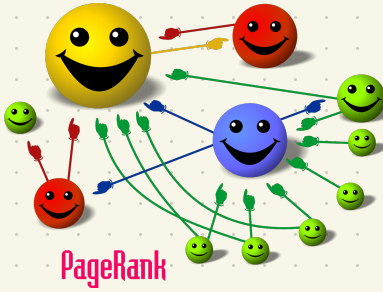
Completing tasks
with deadlines

KILLER APPS

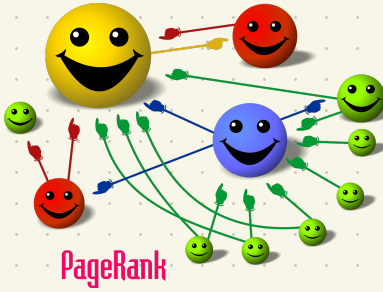
KILLER APPS



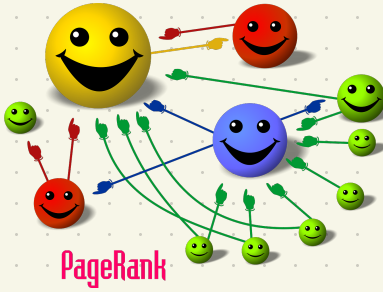
KILLER APPS



KILLER APPS



KILLER APPS



WHY SHOULD YOU STUDY ALGORITHMS ?

WHY SHOULD YOU STUDY ALGORITHMS?

* Important for all branches of computer science

Networks

shortest-path
algorithms

Cryptography

number-theoretic
algorithms

Graphics

geometric
algorithms

Databases

search tree
data structures

WHY SHOULD YOU STUDY ALGORITHMS?

* Important for all branches of computer science

Networks

shortest-path
algorithms

Cryptography

number-theoretic
algorithms

Graphics

geometric
algorithms

Databases

search tree
data structures

* A "lens" for non-CS areas

Physics

quantum computation

Economics

understanding "equilibria"

Biology

evolution as a search algorithm

WHY SHOULD YOU STUDY ALGORITHMS?

* Important for all branches of computer science

Networks

shortest-path
algorithms

Cryptography

number-theoretic
algorithms

Graphics

geometric
algorithms

Databases

search tree
data structures

* A "lens" for non-CS areas

Physics

quantum computation

Economics

understanding "equilibria"

Biology

evolution as a search algorithm

* Challenging puzzles — you will feel smarter!

WHY SHOULD YOU STUDY ALGORITHMS?

* Important for all branches of computer science

Networks

shortest-path
algorithms

Cryptography

number-theoretic
algorithms

Graphics

geometric
algorithms

Databases

search tree
data structures

* A "lens" for non-CS areas

Physics

quantum computation

Economics

understanding "equilibria"

Biology

evolution as a search algorithm

* Challenging puzzles — you will feel smarter!

* Fun!

CAKE - CUTTING

What is a fair way to divide
a cake between two people?



CAKE - CUTTING



What is a fair way to divide a cake between two people?

Person 1 cuts the cake into two parts they consider equal in value.

Person 2 chooses their favorite part and Person 1 gets the leftovers.

INTEGER ADDITION

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

$$\begin{array}{rcccc} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \\ \hline \\ \hline \end{array}$$

Grade-school addition

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

$$\begin{array}{rcccc} & & & & 1 \\ & 5 & 6 & 7 & 8 \\ & 1 & 2 & 3 & 4 \\ \hline & & & & 2 \\ \hline \end{array}$$

Grade-school addition

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
		1	2

Grade-school addition

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
	9	1	2

Grade-school addition

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

Grade-school addition

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

Grade-school addition

This algorithm is correct.

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

Grade-school addition

This algorithm is **correct**.

How much **time** does it take?

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

basic operation

adding two single-digit numbers

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

basic operation

adding two single-digit numbers

n single-digit addition operations

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

basic operation

adding two single-digit numbers

n single-digit addition operations

$n-1$ carry operations (at most)

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

basic operation

adding two single-digit numbers

n single-digit addition operations

$n-1$ carry operations (at most)

adding a 2-digit and a 1-digit number

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

e.g.,

5	6	7	8
1	2	3	4
6	9	1	2

basic operation

adding two single-digit numbers

n single-digit addition operations

$\begin{cases} n-1 \text{ carry operations (at most)} \\ \text{adding a 2-digit and a 1-digit number} \end{cases}$
 $\leq 2(n-1)$ single-digit addition operations

INTEGER ADDITION

input: two n digit numbers x and y

output: the sum $x + y$

eg.,

5	6	7	8
1	2	3	4
6	9	1	2

$\leq 3n$ operations overall

basic operation

adding two single-digit numbers

n single-digit addition operations

$\left\{ \begin{array}{l} n-1 \text{ carry operations (at most)} \\ \text{adding a 2-digit and a 1-digit number} \end{array} \right.$

$\leq 2(n-1)$ single-digit addition operations

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5	6	7	8
1	2	3	4

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} 5678 \\ 1234 \\ \hline 22712 \end{array}$$

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

eg.,

5	6	7	8	
1	2	3	4	
2	2	7	1	2
1	7	0	3	4

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

eg.,

		5	6	7	8	
		1	2	3	4	
		2	2	7	1	2
		1	7	0	3	4
1	1	3	5	6		

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

		5	6	7	8	
		1	2	3	4	
		2	2	7	1	2
	1	7	0	3	4	
1	1	3	5	6		
5	6	7	8			

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5 6 7 8

1 2 3 4

2 2 7 1 2

1 7 0 3 4

1 1 3 5 6

5 6 7 8

7 0 0 6 6 5 2

Grade-school algorithm

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5 6 7 8

1 2 3 4

2 2 7 1 2

1 7 0 3 4

1 1 3 5 6

5 6 7 8

7 0 0 6 6 5 2

Grade-school algorithm

This algorithm is correct.

How much **time** does it take?

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} \begin{array}{cccc} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \end{array} \\ \hline \begin{array}{cccccc} 2 & 2 & 7 & 1 & 2 \\ 1 & 7 & 0 & 3 & 4 \\ 1 & 1 & 3 & 5 & 6 \\ 5 & 6 & 7 & 8 \end{array} \\ \hline 7 & 0 & 0 & 6 & 6 & 5 & 2 \end{array}$$

basic operations: add or multiply
two single-digit numbers

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} \begin{array}{cccc} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \end{array} \\ \hline \begin{array}{cccccc} 2 & 2 & 7 & 1 & 2 & \\ 1 & 7 & 0 & 3 & 4 & \\ 1 & 1 & 3 & 5 & 6 & \\ 5 & 6 & 7 & 8 & & \end{array} \\ \hline 7 & 0 & 0 & 6 & 6 & 5 & 2 \end{array}$$

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations $\left\{ \begin{array}{l} n \text{ multiplications} \\ \leq 2n \text{ additions} \\ \text{(including carry)} \end{array} \right.$

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} 5678 \\ 1234 \\ \hline \end{array}$$

22712

$$\begin{array}{r} 17034 \\ 11356 \\ 5678 \\ \hline \end{array}$$

11356

5678

7006652

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} \begin{array}{cccc} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \end{array} \\ \hline \begin{array}{cccccc} 2 & 2 & 7 & 1 & 2 & \\ 1 & 7 & 0 & 3 & 4 & \\ 1 & 1 & 3 & 5 & 6 & \end{array} \\ \begin{array}{cccc} 5 & 6 & 7 & 8 \end{array} \\ \hline \begin{array}{cccccc} 7 & 0 & 0 & 6 & 6 & 5 & 2 \end{array} \end{array}$$

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

$$\begin{array}{r} \text{5 6 7 8} \\ \text{1 2 3 4} \\ \hline 22712 \\ 17034 \\ 11356 \end{array}$$

5 6 7 8

7 0 0 6 6 5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

	5	6	7	8	
	1	2	3	4	
	2	2	7	1	2
	1	7	0	3	4
1	1	3	5	6	
5	6	7	8		
7	0	0	6	6	5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations per row

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5 6 7 8
1 2 3 4

2 2 7 1 2
1 7 0 3 4
1 1 3 5 6
5 6 7 8

7 0 0 6 6 5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations per row

n rows

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5 6 7 8
1 2 3 4

2 2 7 1 2
1 7 0 3 4
1 1 3 5 6
5 6 7 8

7 0 0 6 6 5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations per row

n rows

?

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5 6 7 8
1 2 3 4

2 2 7 1 2
1 7 0 3 4
1 1 3 5 6
5 6 7 8

7 0 0 6 6 5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations per row

n rows

$\leq 6n^2$ operations (Exercise)

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

5	6	7	8
1	2	3	4

	2	2	7	1	2
	1	7	0	3	4
1	1	3	5	6	
5	6	7	8		

7 0 0 6 6 5 2

basic operations: add or multiply
two single-digit numbers

$\leq 3n$ operations per row

n rows

$\leq 9n^2$ operations overall

$\leq 6n^2$ operations (Exercise)

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

e.g.,

		5	6	7	8	
		1	2	3	4	
		2	2	7	1	2
		1	7	0	3	4
	1	1	3	5	6	
	5	6	7	8		
	7	0	0	6	6	5 2

High-level message

$$\# \text{ operations} \leq \underbrace{\text{constant}}_{\leq 9} \cdot n^2$$

THE DESIGN AND ANALYSIS OF COMPUTER ALGORITHMS

Alfred V. Aho
Bell Laboratories

John E. Hopcroft
Cornell University

Jeffrey D. Ullman
Princeton University

from selecting an appropriate representation for a problem to performing operations in a judicious order. Perhaps the most important principle for the good algorithm designer is to refuse to be content. The designer should continue to examine a problem from a number of viewpoints until he is convinced that he has the most suitable algorithm for his needs.

CAN WE DO BETTER?

from selecting an appropriate representation for a problem to performing operations in a judicious order. Perhaps the most important principle for the good algorithm designer is to refuse to be content. The designer should continue to examine a problem from a number of viewpoints until he is convinced that he has the most suitable algorithm for his needs.

INTEGER MULTIPLICATION

input: two n digit numbers x and y

output: the product $x \cdot y$

eg.,

		5	6	7	8	
		1	2	3	4	
		2	2	7	1	2
	1	7	0	3	4	
1	1	3	5	6		
5	6	7	8			
7	0	0	6	6	5	2

High-level message

$$\# \text{ operations} \leq \underbrace{\text{constant}}_{\leq 9} \cdot n^2$$

Can we do better?

The algorithm design space is surprisingly rich !

KARATSUBA MULTIPLICATION

KARATSUBA MULTIPLICATION

$$x = 5 \quad 6 \quad 7 \quad 8$$

$$y = 1 \quad 2 \quad 3 \quad 4$$

KARATSUBA MULTIPLICATION

$$\begin{array}{rcl} x = & \overset{a}{\textcircled{5 \quad 6}} & \overset{b}{\textcircled{7 \quad 8}} \\ y = & \underset{c}{\textcircled{1 \quad 2}} & \underset{d}{\textcircled{3 \quad 4}} \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

$$\begin{array}{cc} & a & & b \\ x = & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} \\ & & & & \\ y = & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \\ & c & & d \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

$$\begin{array}{cc} & a & & b \\ x = & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} \\ y = & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \\ & c & & d \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

$$\begin{array}{cc} & a & & b \\ x = & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} \\ & c & & d \\ y = & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

$$\begin{array}{cc} & a & & b \\ x = & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} \\ & c & & d \\ y = & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ compute 6720000 from ①

$$\begin{array}{cc} & a & & b \\ x = & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} \\ & & & & \\ y = & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} \\ & c & & d \end{array}$$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ compute

6	7	2	0	0	0	0	from ①
			2	6	5	2	from ②

$x = \begin{matrix} & a & \\ \text{5} & 6 & \\ & c & \end{matrix} \quad \begin{matrix} & b & \\ \text{7} & 8 & \\ & d & \end{matrix}$

$y = \begin{matrix} & & \\ \text{1} & 2 & \\ & c & \end{matrix} \quad \begin{matrix} & & \\ \text{3} & 4 & \\ & d & \end{matrix}$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ Compute

672	0000	from ①
2652		from ②
2840	00	from ④

$x = \begin{matrix} & a & \\ \text{5} & 6 & \\ & c & \end{matrix} \quad \begin{matrix} & b & \\ \text{7} & 8 & \\ & d & \end{matrix}$

$y = \begin{matrix} \text{1} & 2 & \\ & c & \end{matrix} \quad \begin{matrix} \text{3} & 4 & \\ & d & \end{matrix}$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ Compute

672	0000	from ①
2652		from ②
2840	00	from ④

$x = \begin{matrix} & a & \\ \text{5} & 6 & \\ & c & \end{matrix} \quad \begin{matrix} & b & \\ \text{7} & 8 & \\ & d & \end{matrix}$

$y = \begin{matrix} & & \\ \text{1} & 2 & \\ & c & \end{matrix} \quad \begin{matrix} & & \\ \text{3} & 4 & \\ & d & \end{matrix}$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ Compute

672	0000	from ①
2652		from ②
2840	00	from ④
7006652		

$x = \begin{matrix} & a & \\ \text{5} & 6 & \\ & c & \end{matrix} \quad \begin{matrix} & b & \\ \text{7} & 8 & \\ & d & \end{matrix}$
 $y = \begin{matrix} \text{1} & 2 & \\ & c & \end{matrix} \quad \begin{matrix} \text{3} & 4 & \\ & d & \end{matrix}$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

④ compute ③ - ② - ① = 2840

⑤ Compute

672	0000	from ①
2652		from ②
284000		from ④
7006652		= $x \cdot y$

$x = \begin{matrix} & a & \\ \text{5} & 6 & \\ & c & \end{matrix} \quad \begin{matrix} & b & \\ \text{7} & 8 & \\ & d & \end{matrix}$
 $y = \begin{matrix} & & \\ \text{1} & 2 & \\ & c & \end{matrix} \quad \begin{matrix} & & \\ \text{3} & 4 & \\ & d & \end{matrix}$

KARATSUBA MULTIPLICATION

① compute $a \cdot c = 56 \times 12 = 672$

② compute $b \cdot d = 78 \times 34 = 2652$

③ compute $(a+b) \cdot (c+d) = 134 \times 46 = 6164$

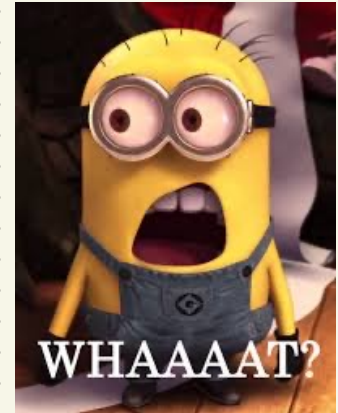
④ compute $\textcircled{3} - \textcircled{2} - \textcircled{1} = 2840$

⑤ Compute

$$\begin{array}{r}
 6720000 \quad \text{from } \textcircled{1} \\
 2652 \quad \text{from } \textcircled{2} \\
 284000 \quad \text{from } \textcircled{4} \\
 \hline
 7006652 = x \cdot y
 \end{array}$$

$$\begin{array}{cc}
 a & b \\
 \hline
 5 & 6 & 7 & 8 \\
 \hline
 1 & 2 & 3 & 4 \\
 \hline
 c & d
 \end{array}$$

$x =$ (5 6) (7 8)
 $y =$ (1 2) (3 4)



Designing fast algorithms

Divide-and-conquer

Graph algorithms

Greedy algorithms

Dynamic programming

Network flow

Linear programming

Designing fast algorithms

Divide-and-conquer

Graph algorithms

Greedy algorithms

Dynamic programming

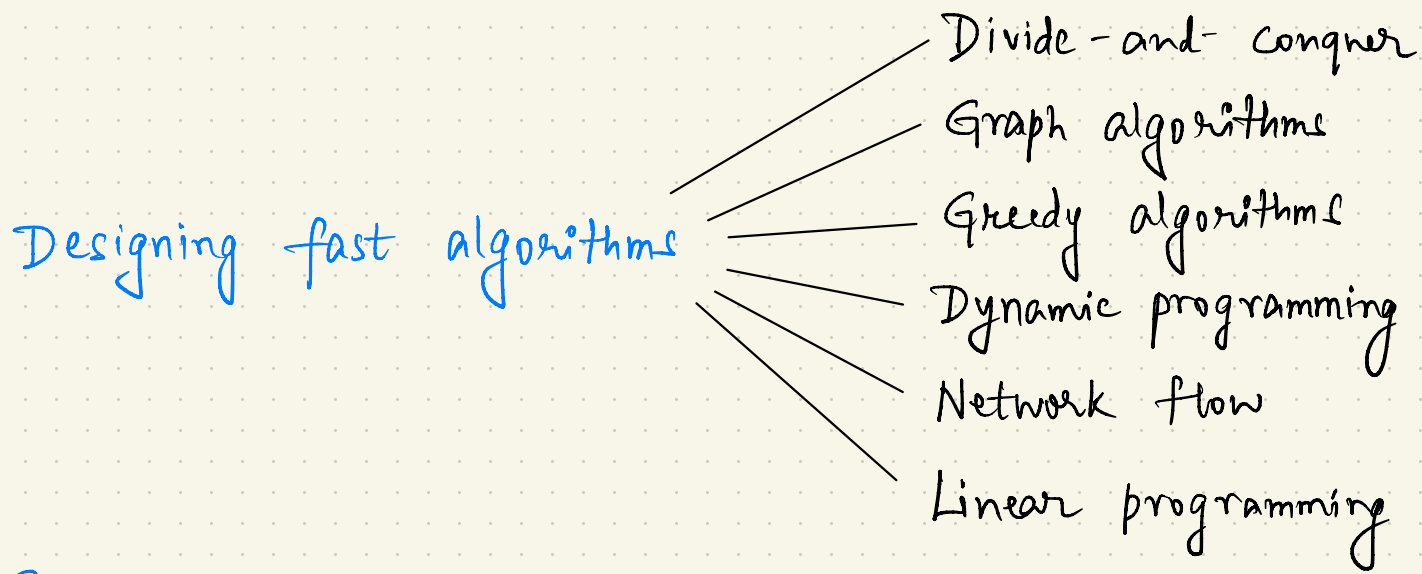
Network flow

Linear programming

Proving that a fast algorithm
probably doesn't exist

NP-completeness

Designing fast algorithms



Proving that a fast algorithm
probably doesn't exist

NP-completeness

Coping with NP-completeness

Approximation
Randomization
Parameterization

COURSE LOGISTICS

INSTRUCTOR

ROHIT VAISH (Please call me Rohit)

Lectures: Tue / Wed / Fri, 10 AM, LH 410

COURSE WEBPAGE



TEAMS CHANNEL



TEACHING ASSISTANTS

ABHISHEK TIWARI

OMAR ISLAM LASKAR

ANCHUL VIJAYVERGIYA

SHARAYU DESHMUKH

NAYAN RABADIYA

SATWIK

YASH BANSAL

SANCHITA SAHA

KALAVALA NAVYA

TEACHING ASSISTANTS

ABHISHEK TIWARI

OMAR ISLAM LASKAR

ANCHUL VIJAYVERGIYA

SHARAYU DESHMUKH

NAYAN RABADIYA

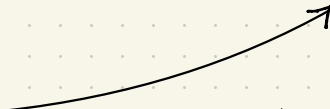
SATWIK

YASH BANSAI

SANCHITA SAHA

KALAVALA NAVYA

Tutorials on Mon/Tue/Thurs/Fri, 1-2 PM, LH 313.5



TEACHING ASSISTANTS

ABHISHEK TIWARI

ANCHUL VIJAYVERGIYA

NAYAN RABADIYA

YASH BANSAI

KALAVALA NAVYA

OMAR ISLAM LASKAR — Group 1, Monday

SHARAYU DESHMUKH — Group 2, Tuesday

SATWIK — Group 3, Thursday

SANCHITA SAHA — Group 4, Friday

Tutorials on Mon/Tue/Thurs/Fri, 1-2 PM, LH 313.5

(check your group on eAdmin)

TEACHING ASSISTANTS

ABHISHEK TIWARI

ANCHUL VIJAYVERGIYA

NAYAN RABADIYA

YASH BANSAL

KALAVALA NAVYA

OMAR ISLAM LASKAR — Group 1, Monday

SHARAYU DESHMUKH — Group 2, Tuesday

SATWIK — Group 3, Thursday

SANCHITA SAHA — Group 4, Friday

Tutorials on Mon/Tue/Thurs/Fri, 1-2 PM, LH 313.5

(check your **group** on eAdmin)

Office hour on Tue, 6-7 PM, Bharti 408 (Crypto Lab)

EVALUATION

TUTORIAL QUIZZES 30%

IN-CLASS QUIZZES 45%

ASSIGNMENTS 25%

EVALUATION

~ 10 mins	TUTORIAL QUIZZES	30%	[3% x best 10 out of 14]
	IN-CLASS QUIZZES	45%	
	ASSIGNMENTS	25%	

EVALUATION

~ 10 mins	TUTORIAL QUIZZES	30%	[3% x best 10 out of 14]
~ 50 mins	IN-CLASS QUIZZES	45%	[15% x best 3 out of 4]
	ASSIGNMENTS	25%	

EVALUATION

~ 10 mins	TUTORIAL QUIZZES	30 %	[3 % x best 10 out of 14]
~ 50 mins	IN-CLASS QUIZZES	45 %	[15 % x best 3 out of 4]
~ 4 hrs	ASSIGNMENTS	25 %	[25 % x best 1 out of 2]

EVALUATION

~ 10 mins	TUTORIAL QUIZZES	30 %	[3 % x best 10 out of 14]
~ 50 mins	IN-CLASS QUIZZES	45 %	[15 % x best 3 out of 4]
~ 4 hrs	ASSIGNMENTS	25 %	[25 % x best 1 out of 2]

AUDITS

Not allowed

ATTENDANCE

Based on tutorial quizzes

RE-EVALUATION

None

TUTORIALS

TUTORIALS

* Once almost every week

TUTORIALS

* Once almost every week

* Tutorial sheet announced a week in advance

...	M	Tu	W	Th	Fr	Sat	Sun	M	Tu	W	Th	Fr	...
								G1	G2		G3	G4	
	Tutorial sheet Announced												

TUTORIALS

- * Once almost every week
- * Tutorial sheet announced a week in advance

... M Tu W Th Fr Sat Sun M Tu W Th Fr ...
Tutorial sheet G1 G2 G3 G4
Announced

Typical tutorial sheet

~ 6 regular problems

- one of these tested in tutorial quiz (announced on the spot)

~ 2 star(★) problems

- not tested in tutorial quizzes
- useful for other evaluations (in-class quizzes, assignments)

REGRADING

- * Gradescope : Sign up **only** with @ dept.iitd, **not** @ iitd
- * "Frivolous" requests : **-2%** in overall score for first one
Subsequently **-4%** , **-8%** , **-16%** , ...

If in doubt , talk to your group's TA.

PLAGIARISM

In any form, at any stage

Grade change and disciplinary action at instructor's discretion

NO GARBAGE

(Credit : Allan Bonodin, Nisarg Shah)

NO GARBAGE

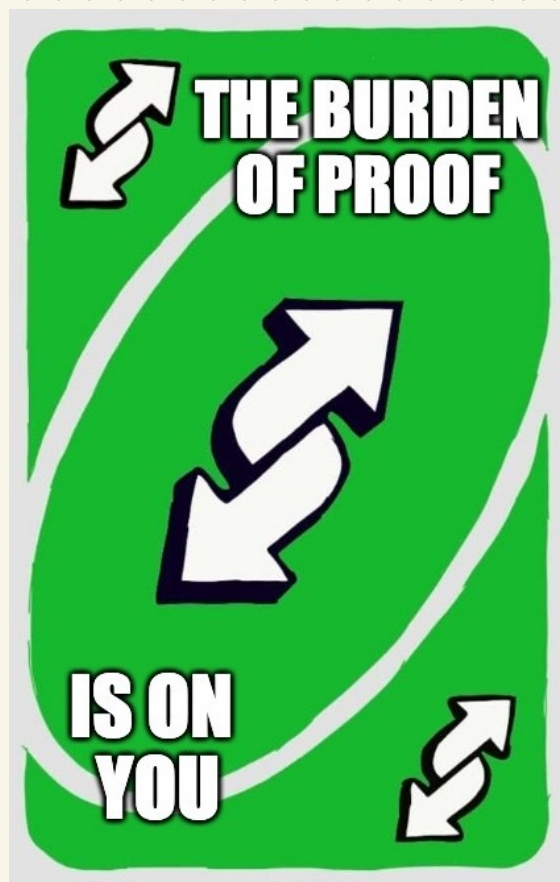
Partial marks for promising approaches [can be extended to give correct answer]

Zero marks if answer makes no sense or is hard to read

20% marks for "I DON'T KNOW".

NOTE : $20 \geq 0$

(Credit : Allan Borodin, Nisarg Shah)



COMMUNICATION

* Most important!

* Please reach out to me or TAs about ANY issues at any time. Do not wait until the end of semester.

* We want you to enjoy learning about algorithms and do well in the course.

All the best!