COL 3	51 : ANALYCIS	& DESIGN OF	ALGORITHMS	
· · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · ·	· · · · · · ·
· · · · · · · · · · · · · · · ·	1507	T110E 20		· · · · · · ·
· · · · · · · · · · · · · · · ·				· · · · · · ·
NP-	COMPLETENE	SS I : INTR	ODUCTION	· · · · · · ·
· · · · · · · · · · · · · · · · · · ·		· · · · · · · · · · · · · · · ·		· · · · · · ·
· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · ·	· · · · · · · · · · · · · · ·	· · · · · · ·
N	OV 05 2024	I ROHI	H2HAV T	· · · · · · ·

	1	•		•			•							1	• •			•											•	• •			
											• 6	<b>.</b> –	••	Λ.			1		-	0													
												>	U	K	1.	2	20		1	łK													
		• •		• •								• •			•															• •			
		• •		• •								• •						• •												• •			
		• •		• •								• •						• •												• •			
		• •		• •								• •						• •												• •			
										• •								• •											•				
										• •								• •											•				
		• •		• •						• •					• •			• •					• •			•	• •		•				
		• •		• •						• •					• •			• •					• •			•	• •		•				
		• •		• •						• •					• •			• •					• •			•	• •		•				
		• •		• •			• •			• •		• •			• •			• •			• •		• •	•			• •	•	•			•	
	•	• •		• •			• •	•		• •		• •		•	• •		•	• •			• •		• •	•		•	• •		•	• •		•	
	•	• •		• •			• •	•		• •		• •		•	• •		•	• •			• •		• •	•		•	• •		•	• •		•	
							• •	•		• •		• •			• •			• •						•		•	• •		•				•
							• •	•		• •		• •			• •			• •						•		•	• •		•				•
							• •	•		• •		• •			• •			• •						•		•	• •		•				•
							• •	•		• •		• •			• •			• •						•		•	• •		•				•
							• •	•		• •		• •			• •			• •						•		•	• •		•				•
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	•
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
	•	• •		• •			• •			• •		• •		•	• •			• •			• •		• •				• •	•				•	
							• •	•		• •		• •			• •			• •						•		•	• •		•			•	•
							• •	•		• •		• •			• •			• •						•		•	• •		•			•	•
		• •		• •		•	• •			• •		• •			• •			• •			• •		• •	•		•	• •		•			•	
		• •	•	• •		•	• •			•		• •			• •			• •			• •		• •	•		•	• •	•	•			•	
		• •		• •			• •			• •		• •			• •			• •			• •		• •	•		•	• •	•	•		•	•	

	STORY SO FAR	· · · · · · · · · · · · · · · · · · ·
Problem	Algorithm	Running time
Sorting	Merge Sort	O (n log n)
Strongly Connected Components	Kosanaju	0(n+m)
Shortest Paths	Dijkstna	$O((m+n) \log n)$
Minimum Spanning Tree	Kruskal	0 (m leg n)
Sequence Alignment	Needleman-Wunsch	0 (mn)
All Pains Shortest Paths	Floyd-Waishall	$O(n^3)$

																														•		*	• •				
									. /		•		•												• • • •												
									. <b>f</b>	11			A	L	51	) (	U	Т	H	M		<b>C</b> .		M.	ZY	<b>.</b> T(	E	(Y									
											Ϊ.				7		7		. •	• •	•	Ť			•			1									
		•																																			
	•						• •		• •	•	•		• •							• •					•	•	• •		•	• •		•					
	•		• •			•	• •		• •				• •		•					• •					•	•	• •			• •	•						
			• •																										•			•					
																									•												
																									•												
	•		• •		•		• •		• •		•		• •		•					• •						•	• •			• •		•					
			• •				• •		• •		•		• •		•					• •					•		• •			• •		•	• •			• •	
			•			•																			•								• •			• •	
	•		• •			•	• •		• •	•	•	•	• •		•				•	• •				•	•	•	• •	•	•	• •		•	• •			• •	
			• •			•			• •															• •		•	• •		•	• •		•			•	• •	*
	•		• •			•	• •		• •		•	•	• •		•				•	• •				• •	•	•	• •	•		• •		•	• •			• •	•
			• •			•	• •		• •			•	• •		•				•	• •			•	• •		•	• •			• •		•	• •				•
			• •			•	• •		•				• •		•					• •				•			•			• •		•	• •		1	• •	•
	•		• •			•	• •		• •			•	• •		•				•	• •				• •		•	• •			• •		•	• •		•	• •	
			• •			•	• •		•				• •		•					• •				•		•	• •			• •		•	• •	•	•		
			• •			•	• •		• •				• •		•					• •				• •			• •			• •		•	• •	•	•	• •	•
			• •				• •		• •				• •							• •				• •			• •			• •					•		



· · · · ·	AN	ALGORITHMIC	MYSTERY	· · · · · · · · · · · · · · · · · · ·
  	Minimum Spanning To	ree	· · · · · · · · · ·	
input:	an undirected graph G	=(V, É)	· · · · · · · · · ·	
· · · · ·	with heal-valued edge	cost Ce	· · · · · · · · · ·	
· · · · ·	Tor each eage e			· · · · · · · · · · · · · · · · · · ·
· · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · ·	· · · · · · · · · · · · · · · · · ·
· · · · ·	· · · · · · · · · · · · · · · · · · ·			· · · · · · · · · · · · · · · · · · ·
· · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · ·	
· · · · ·	· · · · · · · · · · · · · · · · · · ·			· · · · · · · · · · · · · · · · · · ·
			· · · · · · · · ·	

AN ALGORITHMIC	Mystery
Minimum Spanning Thee	
input: an undirected graph G=(V,E)	
with heal-valued edge cost Ce	
for each edge e	
ontput: a spanning tree T with	
minimum total cost 2 Ce. eET	

AN ALGORITHMIC	L MYSTERY
Minimum Spanning Tree	
input: an undirected graph G = (V, E)	
with heal-valued edge cost Ce	
Outbut: a spanning tree T with	
minimum total cost Z Ce.	
eeT search space : n shanning trees in	
Complete graphs	

AN ALGORITHMIC	. Mystery
Minimum Spanning Tree	
input: an undirected graph G=(V,E)	
with heal-valued edge cost Ce for each edge p	
output: a spanning tree T with	
minimum total cost Z Ce.	
search space : n = 2 spanning trees in	
Complete graphs	
algorithms: Phim's, Kauskal's, Ofm log n)	

AN ALGORITHMIC	MYSTERY		· · · · · · · ·
Minimum Spanning Thee	Traveling	Salisman	Problem
input: an undirected graph G=(V,E)			· · · · · · · · ·
with heal-valued edge cost G			
for each edge e			
output: a spanning tree T with	· · · · · · · · · · · ·	· · · · · · ·	· · · · · · · · ·
minimum total cost Z Ce. eET	· · · · · · · · · · · · ·	· · · · · · ·	· · · · · · · · · ·
Search space: n = spanning trees in	· · · · · · · · · · · · ·		 
Complete graphs			· · · · · · · ·
algorithms: Phim's, Kauskal's, O(m log n)	· · · · · · · · · · · ·	· · · · · · ·	· · · · · · · · ·

AN ALGORITHI	MIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undirected graph G=(V,E)	input: an undirected graph $G = (V, E)$
with heal-valued edge cost Ce	with heal-valued edge cost Ce
for each edge e	for each edge e
ontput: a spanning tree T with	
minimum total cost Z Ce. eET	
search space: n <sup>-2</sup> spanning trees in	
Complete graphs	· · · · · · · · · · · · · · · · · · ·
algorithms: Phim's, Kauskal's, O(m log n)	

AN ALGORIT	HMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undirected graph G=(V,E)	input: an undirected graph G=(V,E)
with heal-valued edge cost Co	with heal-valued edge cost Co
for each edge e	for each edge e
output: a spanning tree T with	ontput: a tour T with
minimum total cost 2 ce. eET	minimum total cost Z Ce. eET
search space: n spanning trees in	
Complete graphs	· · · · · · · · · · · · · · · · · · ·

AN ALGORITH	IMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undinected graph G=(V,E)	input: an undirected graph G=(V,E)
with heal-valued edge cost Ce	with heal-valued edge cost Ce
for each edge e	for each edge e
Output: a spanning tree T with	ontput: a tour T with
minimum total cost Z Ce. eET	minimum total cost Z Ce. CET
Search space : n <sup>n-2</sup> spanning trees in Complete graphs	
algorithms: Phim's, Kanskal's, Ofm log n)	

AN ALGORIT	HMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undirected graph $G = (V, E)$	input: an undirected graph G=(V,E)
with heal-valued edge cost Ce	with heal-valued edge cost Ce
for each edge e	for each edge e
output: a spanning tree T with	ontput: a tour T with
minimum total cost 2 ce. eET	minimum total cost Z Ce. eET
search space: n <sup>-2</sup> spanning trees in	
Complete graphs	
algorithms: Phim's, Kauskal's, O(m log n)	

AN ALGORITH	IMIC MYSTERY
Minimum Spanning Tree	Traveling Salisman Problem
input: an undirected graph G=(V,E)	input: an undirected graph $G = (V, E)$
with heal-valued edge cost Ce	with heal-valued edge cost Ce
for each edge e	for each edge e
output: a spanning tree T with	ontput: a tour T with
minimum total cost Z Ce. eET	minimum total cost Z Ce. eET
Search space : n <sup>n-2</sup> spanning trees in Complete graphs	
algorithms: Phim's, Kanskal's, O(m log n)	

AN ALGORIT	HMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undirected graph G=(V,E)	input: an undirected graph G=(V,E)
with heal-valued edge cost Co	with heal-valued edge cost Co
for each edge e	for each edge e
output : a spanning tree T with	ontput: a tour T with
minimum total cost Z Ce. eET	minimum total cost Z Ce. eET
Search space: n spanning trees in	search space: ?
complete graphs	
algorithms: Prim's, Kanskal's, O(m log n)	

AN ALGORIT	HMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undinected graph G=(V,E)	input: an undirected graph G=(V,E)
with heal-valued edge cost Ce	with neal-valued edge cost Ce
for each edge e	for each edge e
Output: a spanning tree T with	ontput: a tour T with
minimum total cost Z Ce. eET	minimum total cost Z Ce. eET
Search space : n <sup>n-2</sup> spanning trees in Complete graphs	search space: $\frac{1}{2}(n-1)!$ on complete graphs
algorithms: Prim's, Kruskal's, Ofm log n)	

AN ALGORIT	HMIC MYSTERY
Minimum Spanning Thee	Traveling Salisman Problem
input: an undinected graph G=(V,E)	input: an undirected graph G=(V,E)
with heal-valued edge cost Ce	with heal-valued edge cost Ce
for each edge e	for each edge e
Output: a spanning tree T with	ontput: a tour T with
minimum total cost Z Ce. CET	minimum total cost Z Ce. eET
Search space : n <sup>-2</sup> spanning trees in Complete graphs	search space: $\frac{1}{2}(n-1)!$ on complete graphs
algonithms: Prim's, Knuskal's, O(m log n)	algorithms :

· · ·	AN ALGO	RITHMIC MYSTERY
	.pplications	Traveling Salisman Problem
×	sequenced processing of tasks	input: an undirected graph $G = (V, E)$
,	genome he construction	with heal-valued edge cost Ce
· · · ·	many more	for each edge e
	<u>d</u>	ontput: a tour T with
· · · ·		minimum total cost Z Ce. eET
· · · ·		search space: $\frac{1}{2}(n-1)!$ on complete graphs
· · ·		algorithms :

AN ALGORIT	HMIC MYSTERY
Applications	Traveling Salusman Problem
* sequenced processing of tasks	input: an undirected graph G = (V, E)
* genome reconstruction	with heal-valued edge cost Ce
t many more	for each edge e
d.	ontput: a tour T with
Fact	minimum total cost Z Ce.
No known fast (i.e., poly time)	eet.
algorithm for TSP	search space: $\frac{1}{2}(n-1)!$ on complete graphs
(despite decades of effort)	algonithms :



Source: Garry and Johnson (1979)



JOURNAL OF RESEARCH of the National Bureau of Standards—B. Mathematics and Mathematical Physics Vol. 71B, No. 4, October—December 1967	, , , ,		•	•	•	· ·
<b>Optimum Branchings*</b>	· · ·	· ·	•	•	•	· ·
Jack Edmonds	) 0 ) 0	) 0 0 0	•	•	•	• •
			•	•	•	• •
traveling saleman problem [cf. 4]. I conjecture that there is no good algorithm for the traveling saleman			•	•	•	• •
matical conjecture: (1) It is a legitimate mathematical possibility, and (2) I do not know.			•	•	•	• •
900d = bolunomial - time			•	•	•	• •
Querra d'internet et et			•	•	•	• •

•	•			•	•									•		•	•		•		•	•			•		•	•	•	•	•	•			•			•		•	•	•	•	•				•	•
			•	Sa	).	f	-Ak			W	e	 h	۱۵۰	 n	Ł	- 1	m	G.V	\a	ġ	ed	Ĺ		t	ō	•	b	י ריס'	Vł	).		E	di	m	)M	di	2	•	Ċ	or	١Ì	ec	ħ	M	ο,				
					•	. 1		·).						•••	Č.					Ĩ		-					P				•									•	đ								•
•							• •		•										•	."					•		•	•		•	•	•								•		•	•	•	•	•	•	•	•
•							• •			•				•			•	•	•			•	•	•	•	•	•	•	•	•	•				•			•		•		•	•	•	•	•		•	
							• •							•		•	•							•		•	•	•	•		•				•					•			•	•	•	•			
•	•					•																•	•				•		•		•									•	•	•				•			
						•																									•									•						•			
						•																					•		•		•	•								•	•				•	•		•	•
•						•	• •							•			•		•			•	•	•	•	•	•	•	•	•	•	•			•					•	•	•	•	•		•		•	•
•		•					• •												•			•			•		•	•	•		•	•								•	•	•	•	•	•	•		•	•
•						•	• •						•	•	•		•				•		•		•		•	•	•		•	•			•	•				•	•	•	•	•	•	•			•
						•	• •						•		•			•				•					•		•		•	•				•			•	•	•	•			•	•			
							• •							•		•	•							•		•	•	•	•		•				•					•			•	•	•	•			
						•								•										•		•									•							•				•			
																																														•			
																													•														•						
					•	•	• •											•		•	•	•		•	•	•	•	•	•	•	•	•						•	•	•	•	•	•	•	•	•		•	•
•			•		•	•	• •							•				•		•	•	•	•	•	•	•	•	•	•	•	•	•			•			•	•	•	•	•	•	•	•	•		•	•

••••	• •	So fai	, we ha	venit me	anaged	to prove Ed	monde conjecture.	· · ·
· ·	• •	We	have ide	ntified	thousand	s of TSP-li	ike problems:	· · · ·
· ·	*	Graph	coloring	· · · · · · · · · · · · · · · · · · ·	Intigu	Programming	* Ising Models	· · ·
· ·	×	Longert	Paths	*	Protein	Folding	* Rubik's cube	· · · ·
· ·	*	Number	Partitioning		Boolean	Satisfiability	*	· · ·
	• •	· · · · · · ·	· · · · · · · ·	· · · · · ·	· · · · · · ·	· · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · ·
• •	· ·				· · · · · · ·			· · ·
	• •	· · · · · · ·	· · · · · · · ·	· · · · · ·	· · · · · · ·	· · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · ·
• •								• • •

So far, we have	n't managed to prove Edm	ronds' conjecture.
he have iden	tified thousands of "TSP-lik	« problems :
* Graph coloring	* Integer Programming	* Ising Models
* Longut Paths	* Protein Folding	* Rubik's cube
* Number Partitioning	* Boolean Satisfiability	*
How can we build	a compelling case that the	2 Conjecture is time?

So far, we hav	en't managed	to prove Edmo	nds' conjecture.
We have iden	ntified thousand	s of "TSP-like	2 problems:
* Graph coloring	* Intigu	Programming	* Ising Models
* Longut Paths	* Protein	Folding	* Rubik's cube
* Number Partitioning	* Boolean	Satisfiability	*
How can we build	a Compelling	case that the	Conjecture is true?
- ``@`-	RELATIVE	INTRACTA B	віцту



"I can't find an efficient algorithm, but neither can all these famous people."

Source: Garry and Johnson (1979)

																	•								•			• •		• •		
											F	A .c	1.		D	) <u>.</u>		į.	r.													
											E	A7	• Y		T.	R	D C	っし	t	M												
						• •								• •			• •															
									• •											• •			• •		• •		•			• •		
									• •											• •			• •		• •		•			• •		
	•								• •														• •		• •							
	•								• •														• •		• •							
	•								• •														• •		• •							
		• •							• •		• •											• •	• •		• •		•			• •		
		• •				• •			• •					• •			• •			• •		• •	• •		• •		•			• •		•
	•	• •			•	• •			• •				•	• •			• •			• •		• •	• •		• •		•			• •		
	•	• •			•	• •			• •				•	• •			• •			• •		• •	• •		• •		•			• •		
	•	• •			•	• •			• •				•	• •			• •			• •		• •	• •		• •		•			• •		
	•	• •			•	• •			• •				•	• •			• •			• •		• •	• •		• •		•			• •		
	•	• •			•	• •			• •				•	• •			• •			• •		• •	• •		• •		•			• •		
		• •	• •		•				• •		• •			• •			• •			• •		• •	• •		• •		•			• •		
	•		• •		•			•	• •		• •		•	• •			• •			• •			• •		• •	•	•		•	• •		
	•	• •	• •			• •			• •		• •			• •			• •			• •			• •		• •		•		•	• •		•
	•	• •	• •			• •			• •		• •			• •			• •			• •			• •		• •		•		•	• •		•
	•	• •	• •			• •			• •		• •			• •			• •			• •			• •		• •		•		•	• •		•
	•	• •	• •		•	• •	•		• •		• •			• •			• •			• •	•	• •	• •		• •	•			•	• •		•
	•	• •	• •			• •			• •		• •			• •			• •			• •			• •		• •		•		•	• •		•
	•	• •	• •			• •			• •		• •			• •			• •			• •			• •		• •		•		•	• •		•
	•		• •		•			•	• •		• •		•	• •			• •			• •			• •		• •	•	•		•	• •		
		• •	• •		•				• •		• •			• •			• •			• •		• •	• •		• •		•			• •		
		• •	• •		•				• •		• •			• •			• •			• •		• •	• •		• •		•			• •		
		• •	• •						•					• •			• •			• •		• •	• •		• •		•			• •		
		• •	• •			• •			• •					• •			• •			• •		• •	• •		• •		•			• •		
		• •	• •			• •			• •					• •			• •			• •		• •	• •		• •		•			• •		

· · · ·	· · ·				Easy	PROBLEM	· · · · · · · · · · · · · · · · · · ·		
· · ·	· ·	· · ·	· · · · ·	· · · · · · · ·	· · · · · · · · · ·		· · · · · · ·	· · · · · · · ·	· · · · · · · · · ·
· · · ·	· ·	A	broble	m m m	bolumomia	- time solviable	ŭ ·	Hang ìn	
• • •	• •				poignomine		~ . <b>·</b> ]	ince is	
• • •	Þ	olyne	mial-	time al	gorithm H	hat connectly sol	ves it	on all	in puts.
· · · ·	Þ Þ	polyne	omial-	time al	gorithm t	hat connectly sol	ves it	on all	in puts.
· · · · · · · · · · · · · · · · · · ·		polyne	omial-	time al	gorithm H	hat connectly sol	vee it	on all	in puts.
<ul> <li></li></ul>	Þ	polyne	mial-	time a	gorithm H	hat connectly sol	ves it	on all	in puts.
<ul> <li>· · · ·</li> <li>· · ·</li></ul>	<b>P</b>	polyne	pmial-	time a	gorithm H	hat connectly sol	ves it	on all	in puts.

· · ·	•	e e e	qn	Ń	al	in	Hey (1	· · ·	•	•	• • • • • •	· · ·	•	•	· · ·	EI	2.P	У	· · ·	P	<b>R</b> (	<b>.</b>	s L	EI	M	•	· · ·	•	•	· · ·	•	•	· · ·	•	•	· ·	•	•	· ·	•	· · ·
· ·	•	•	•	Ą	•	þr	ol	ole	r M	•	Ì	2	•	þ	oly	n	0 <b>m</b>	i Ni 0	L	-	tin	m e	, , ,		301	VA	616		Ĭ	<u>1</u>	+	he he	re	•	2 1	• •	an	•	• •	•	· ·
	0	Jg	6)	ù 1	thr	n	-	for	م	ł	hì	ch	•	th	e	<b>)</b>	Sol	₽	sbl	e	i	nþ	ont		<u>S</u> Ì2	te	· · ·	(fi	)X_	O	<	fì)	Xed	ل : <del>-</del>	tin	1e	b	nd	get	F)	• •
• •	•		<b>,</b>	• •	20	cal	فله	• •	m	nl	h	þli	Ø	£i	vel	ly (1		W	th		ir	ירר	e	-1 V	ng	•	CO	mþ	n	h	tik	)n	d	ŀ	>0"	We	ን አ	•	• •	•	• •
• •	•	•	•	• •	•	•	•	• •	•	•			•	•	• •	Ņ		•	• •	•		• •	•	•	. <b>.</b>			•	•	• •		•	• •		•	• •	•	•		•	• •
• •			•	• •				• •		•			٠	•		٠	•		• •	•		• •	•	•	• •		• •	•		• •			• •	•	•	• •	٠	•	• •	٠	• •
• •	٠		•			•		• •					•	•	• •			•				• •		•	• •		•	٠				•	• •		•	• •	•	•	• •	٠	• •
• •	•	•	•		•	•				•	• •		•	•	• •	•	•	•		•	•	• •	•	•	• •	•		•	•		•	•	• •	•	•	• •					• •
• •	•		•			•		• •						•	• •			•				• •		•	• •		•						• •		•	• •		•		•	• •
• •	•	•	•			•		• •	•	•			•	•	• •			•		•		• •	•	•	• •		•				•	•	• •	•		• •		•	• •	•	• •
						•							•	•		•		•			•								•			•									

EASY PI	ROBLEM
Equivalently	
A problem is polynomial	time solvable if there is an
algorithm for which the solvable	input size (for a fixed time budget)
scales multiplicatively with	increasing computational power.
χιρ <sup>4</sup>	<b>v</b>
14	
8	· · · · · · · · · · · · · · · · · · ·
6 4	· · · · · · · · · · · · · · · · · · ·
0 5 10 15 20	

		· · · · ·	E	A S Y S F	ROBL	EM		• • •			
Equ	ivalently	· · · · · ·	· · · · · ·		· · · · ·		· · · · ·	· · ·	· · · · ·	· · · · · · ·	· · · · ·
· · · · ·	A proble	2í m	polyn	omial -	time	vloz	able	ìf ·	thue	ìs an	
algor	rithm for	which	the	solvable	input	- Size	2 (fo	r a	fixed	time bu	dget)
· · · · · · ·	scales	multipli	catively	with	incre	uing	comp	ntat	ional	power	
· · · · ·	x 10 <sup>4</sup>			η	Fix	Lima	 	· · ·	 		• • • • •
	4	· · · · · ·		L		TIME	brage	t.`L	)onble	Compute	power.
	4					1 Ime	brage	t.`L	)onble	Compute	power
	4 2 0 8 6						brage	<b>F</b> . `L		Compute	power.
	4 2 0 8 6 4 2		100	2 			brage	<b>Γ</b> . `L		Compute	pomer

Easy P	ROBLEM	•
Equivalently	· · · · · · · · · · · · · · · · · · ·	•
A problem is polynomial-	time solvable if there is an	•
algorithm for which the solvable	input size (for a fixed time budget)	•
scales multiplicatively with	increasing computational power	•
$\frac{x^{10}}{14}$	Fix time budget. Double compute power.	•
	Before After	•
8	$100 n^2$	•
4 2 100 n <sup>2</sup>	$2^n$	•

Ε	ASY PROBI	.EM		
Equivalently			· · · · · · · ·	· · · · · · · · · · · ·
A problem is polyn	iomial - time	solvable	if thue	ìc an
algorithm for which the	solvable inpu	t size (for	a fixed -	time budget)
scales multiplicatively	with incre	aing compu	tational f	SOWER.
×10 <sup>4</sup>  4	2 <sup>n</sup> Fix	time budget	Double	compute power.
		Bef	bre f	fter
8	lo	0 n <sup>2</sup> 1,00	0,000	?
4	2 0 n	n		
0 5 10 15 20		· · · · · · · · · ·		· · · · · · · · · · · ·

Easy F	PROBLEM
Equivalently	· · · · · · · · · · · · · · · · · · ·
A problem is polynomial-	time solvable if there is an
algorithm for which the solvable	input size (for a fixed time budget)
scales multiplicatively with	increasing computational power
x10 <sup>4</sup> 14	Fix time budget. Double compute power.
	Before After
8 6	1,000,000 1,414,213
4 2 2	n
0 5 10 15 20	· · · · · · · · · · · · · · · · · · ·

Easy F	PROBLEM
Equivalently	
A problem is polynomial.	time solvable if there is an
algorithm for which the solvable	input size (for a fixed time budget)
scales multiplicatively with	increasing computational power
x <sup>10<sup>4</sup></sup> 14	Fix time budget. Double compute power.
	Before After
86	$100 n^2$ 1,000,000 1,414,213
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	2 <sup>n</sup> 1,000,000 ?

Easy F	PROBLEM
Equivalently	· · · · · · · · · · · · · · · · · · ·
A problem is polynomial.	time solvable if there is an
algorithm for which the solvable	input size (for a fixed time budget)
scales multiplicatively with	increasing computational power.
x <sup>10<sup>4</sup></sup> 14	Fix time budget. Double compute power.
	Before After
8 6 6	$100 n^2$ 1,000,000 1,414,213
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	2 <sup>n</sup> 1,000,000 1,000,001

				• •						• •									• •				• •				• •		• •			•	• •						• •	
															λ.	A -+		. 1	24		^	á 4	-																	
														1	<b>ft</b>	KI	D	. ľ	- F	(0	В	L	eP	1																
							•																	•																
														• •																					•					
														• •																					•					
								•						• •												•						•			•	• •			• •	
	•		•	• •		•	•	•	•					• •				•				•	• •			•	• •	•	• •			•			•	• •		•	• •	
			•	• •				•					•	• •				•				•	• •			•	• •	•	• •						•		•	•	• •	
•			•	• •	•		•	•	•				•	• •				•	• •				• •			•	• •	•	• •			•		•	•	• •	•	•	• •	
•			•	• •	•		•	•	•				•	• •				•	• •				• •			•	• •	•	• •			•		•	•	• •	•	•	• •	
			•	• •			•	•	•		•			•	•			•	• •	•		•	• •		•	•	• •	•	• •	•	•	•		•	•	• •		•	• •	
	•		•	• •	•	•	•	•	•	• •	•	•	•	• •				•	• •	•		•	• •		•	•	• •	•	• •		•	•			•	• •		•	• •	
			•	• •			•	•	•		•			•	•			•	• •	•		•	• •		•	•	• •	•	• •	•	•	•		•	•	• •		•	• •	
		•		• •		•	•	•	•					• •				•				•	• •			•		•	• •		•	• •	• •		•	• •		•	• •	
		•		• •		•	•	•	•					• •				•				•	• •			•		•	• •		•	• •	• •		•	• •		•	• •	
		•		• •		•	•	•	•					• •				•				•	• •			•		•	• •		•	• •	• •		•	• •		•	• •	
		•		• •		•	•	•	•					• •				•				•	• •			•		•	• •		•	• •	• •		•	• •		•	• •	
			•	• •		•	•	•	• •					• •				•				•	• •			•	• •	•	• •			• •			•	• •	•	•	• •	
		•		• •		•	•	•	•					• •				•				•	• •			•		•	• •		•	• •	• •		•	• •		•	• •	
•			•	• •	•		•	•	•	• •			•	• •				•	• •			•	• •			•	• •	•	• •			•	• •		•	• •		•	• •	
•	•		•	• •	•	•	•	•	• •	• •	•	•	•	• •				•	• •	•		•	• •		•	•	• •	•	• •		•	•	• •		•	• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
				• •		•	•		•					•				•				•	• •					•	• •		•	•				• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
	•			• •			•	•	•					•				•				•	• •		•	•	• •	•	• •			•	• •			• •		•	• •	
•			•	• •	•		•	•	•	• •			•	• •				•	• •			•	• •			•	• •	•	• •			•	• •		•	• •		•	• •	
•			•	• •			•	•	•	• •			•	• •				•	• •			•	• •	•		•	• •	•	•			•	•			• •		•	• •	
			•	• •			•	•	•	• •			•	•			•	•	•			•	• •	•		•	• •	•	•			•	• •	1		• •		•	•	
			•	• •			•	•	•	• •				•				•	• •			•	• •			•	• •	•	•			•	•		•	• •		•	•	

•	•	•	•	•	•	0	•	• •	0	0	•	•	•	•	0	•	•	•	H	F		27	Ď	•	P	Ŕ	0	B	L	E	M	ľ	0	0	• •	•		•			•	• •			•	0	•	•	0	• •	
					•																												•																•		
•	•	•	•	•	•	•	•	A	0	þ	ΥŪ	)6	le	s N	ſ	•	ł	ગે	H	١	. 1	n (	) )	•	þ	D	yı	n rc	m	10	l	-	ť	i m	ne	•	с С	lg	10 }-	י ערי י	th	m		2	•	0	•	•			
																											Ņ											Ņ													
												•																																•			•	•			
								• •				•																		•	•				• •			•	•	•		-	• •	•			•	•	•	•	
				•		•	•	• •			•	•	•									•								•	•		•	•	• •	•	•	•	•	•		•	• •	•			•	•	•	•	
					•		•	• •		•	•	•	•	•					•		•	•		•					•	•	•		•	•	• •	•	•		•	•	•	•					•	•	•	•	
		•	•	•			•	• •						•	•	•		•		•	•			•		•	•		•				•	•	• •	•	•			•								•	•		
		•	•		•	•	•																											•					•	•											
				•																		•																													
																																												•				•		•	
							•	• •				•	•																		•				• •							•		•			•	•	•	•	
					•	•	•																								•			•						•		•	• •	•			•	•	•	•	
		•	•				•	• •		•	•	•	•					•	•			•	•		•					•	•				• •					•		•	•	•			•	•	•		
		•	•			•	•	•			•		•		•							•											•	•	•	•	•			•		•						•	•	•	
		•	•		•	•	•	• •			•											•												•	• •				•	•		•						•	•	•	
		•	•	•			•	• •						•	•	•		•		•	•			•		•	•		•				•	•	• •	•	•			•								•	•		
		•	•		•	•	•																											•		•			•	•											

· · ·	•	•	•	· ·	•	•	•	· · ·	•	•	•	· · ·	•	•	•	4,	A	<b>R</b> 1	Ď		2 <b>6</b>	۲ ۲ ۲	B	LE	Er	1	•	· ·	•	· ·	•	•	•	· ·	•	•	· ·	•	•	· ·	•	· ·
· · ·	•	•	•	· · ·	•	Â	ţ	þ	Y	b b	le	m	•		) )	th /		Ύ (	<b>7</b>	•	þo	lyn	י 100 1	mì	al	, , , , , , , , , , , , , , , , , , ,	-ti	m	e			0,-	ù (	'n	n		· · · · · · · · · · · · · · · · · · ·	•	•	· · ·	•	· · ·
· · ·	•	•	be	۵.	u	se		th	U{	2	Ì	(n	ł	0	m	<u>o</u>	? '	•	•	•	· · ·	Or	· · · · · · · · · · · · · · · · · · ·	þ	ور	ñu	se		~ ~و	· · ·	ha	Vea	n'4	Ļ.	fo	144	rd	<b>0</b> 1	ne	Y	مل	7 7
•••	0	•	•	• •	•	•	0	••••	•	•	•	• •	•	•		• •	•	•	•		• •	•	•		• •	•	•	• •	•	• •	•	•	•	• •	•	•	• •	•		••••	•	• •
	•	•	•	• •			•	• •	•		•		•				•			•												•	•		•	•	• •	•				• •
								• •				• •									• •				• •															• •		• •
• •			•	• •			•	• •	•			• •	•	•		• •				•	• •			•	• •	•	•	• •		•			•	• •		•	• •		•	• •	•	• •
• •		•	•	• •	•		•	• •				• •				• •			•	•	• •			•	• •		•	• •	•	• •	•			• •	•	•	• •	•	•	• •	•	• •
			•				•	• •			•	• •			•	• •			•	•	• •				• •		•	• •			•			• •	•			•	•	• •		
												• •																														
												• •																														
• •				• •				• •				• •			•	• •					• •			•	• •			• •		•			•	• •			• •			• •		• •
• •				• •				• •				• •				• •																					• •					

•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		ł	ł/	4	R	D		P	R	0(	B	LE	73	1	•	•	• •		•	•	•	•	•	• •	•	•	•	•	•	•	• •	•	
	•	•	•	•	•	•	•	A	l	•	þ	Υſ		5   e	20		•	•	ر الالا		h -		n	0	•	Þ		yn	101	ni	al		+	Ìn	16			19	0,	Ň	th	m		2 - - -	•	•	•	•	· · ·		•
• • • •	•	•	•	b	ور	- <b>a</b> v	u S	e e n	•	-{1	ht 1	J.(		•	'2í	י מי ה	t		ογ 0γ	16		? •	•		•	•		٦ ۲	•	b	ود	ñ. R.U.	<u>ç</u> e		~ ~	e		     	len Ven	, , '- , ,	ŧ	f	, 00	nd		07 07	ne	J	J	. 7 •	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	he	.'l	L L	r r r	12	V و	م م		 rd	n M	it	•	t		, , , ,	hi	2		•	· ·	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •		•	•	•			•	•	•	•	•	· · ·	•	•	•	•		•	· · ·	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	· ·		•	•	•	•		•	•	•	•	•	· ·	•	•	•	•	•	•	· ·	•	•

· · ·	•	•	•	•	· ·		•	-	•	•	•	•	•	•	•	•	ŀ	1	A	R	t	)	F	<b>2</b> f	20	8	5L	E	M		•	• •	· ·	•	•	•	•	•	•	•	•		· ·	• •		•	•	•	· ·
· · · · · · · · · · · · · · · · · · ·	•	•	•	•	• •		4	•	þ	ΥŪ	р b c	1e	łr	Ň	•	7	n N	h ·		Ŷ	0	• •	· · ·	þo	J	h	0 <b>%</b>				ti	m	e	•		19	0)	Â,	H	יי זי <b>ר</b>		- - - - -	)	· · ·		•	•	•	· · ·
· · · · · · · · · · · · · · · · · · ·	•	•	be	2.04	<b>3</b> 11	se	· · · · · · · · · · · · · · · · · · ·	+	he	Je	· · · · · · · · · · · · · · · · · · ·		(n	ťŧ	•		ne	2	?	· · ·		· · ·	· · ·	· · ·	O	י ה ה ה		oe	<b>CA</b>		ze		2	2		\ <b>A</b> ' 	ve	<b>n'</b>	t	~	foi		J	0	me	, 1   	y e (	₽Ş	)
· · · · · · · · · · · · · · · · · · ·	•	•	•	•	Se	em to	2	þı	ín <sup>.</sup> roi	1 tiv	m	id H	a vi	Fiv s		•	•	•	•	• •		• •	• •			•	h h	10	U	•	'n	٩١	Je	- -		 2d	.m	it		- -		•	th	2 iv		•	•	•	· · ·
· ·	•	•	•	•	• •		•	•	•	•	•	•	•	•	•	•	•	•	•	• •			• •			•	•	•	•	•	•	• • •		•	•	•	•	•	•	•	•	•		• •		•		•	
· ·	•	•	•	•	· ·		•	•	•	•	•	•	•	•	•	•	•	•	•	• •		• •	· ·	· ·	· ·	•	•	•	•	•	•	• •	· ·	•	•	•	•	•	•	•	•	•		· ·		•	•	•	• •

HARD	PROBLEM
A problem with no	polynomial-time algorithm?
because there isn't one?	or because we haven't found one yet?
seems intimidating to prove this	we'll never admit to this!
Can we collect cincumstantial	
evidence to justify this?	

•	• •	· · · · ·		· · · · · · ·	HARD	PROBLEM	1			· · · · · · · ·	· · ·
•	• •	Weak	evidence	of have	lnecs	· · · · · · · · ·	· · · · · ·	· · · ·	· · · · · ·		· · ·
•	• •	· · · · ·		· · · · · · ·	· · · · · · ·	· · · · · · · · · ·	· · · · · ·	· · · · ·	· · · · · ·		· · ·
•	· ·	· · · · ·		· · · · · · · ·	· · · · · · · ·	· · · · · · · · · ·	· · · · · ·	· · · · ·	· · · · · ·		
•	• •	· · · · ·	· · · · · · ·	· · · · · · ·	 	· · · · · · · · ·	· · · · · ·	· · · · ·	· · · · · ·		
•		Strong	evidence	of ha	dness	· · · · · · · · ·	· · · · · ·		· · · · · ·		· · ·
•	• •			· · · · · · · ·	· · · · · · ·	· · · · · · · · · ·	· · · · · ·	· · · ·	· · · · · ·		· · ·
•	· ·	· · · · ·		· · · · · · ·	· · · · · · ·	· · · · · · · · · ·	· · · · · ·		· · · · · ·		· · ·
•	• •	· · · · ·		· · · · · · · ·	· · · · · · · ·	· · · · · · · · · ·	· · · · · ·	· · · · ·	· · · · · ·		· · · ·
•											

· ·	•	•	· ·	•	•	• •	•	•	•	• •	0	•	•	· ·	H	A	R	D	•	P	RO	B	L	Er	1	•	••••	•	· ·	•	•	•	· ·	•	•	· ·		•	· ·	•	•	•
• •	•	ω	ea	K	E	<u>evì</u>	den	nC	e	0	f	ł	NA)	nd	٨e	22	• •		•	•	• •	•	•	• •	•	•	•	•	• •	•	•	•	• •	•	•	• •	•	•	•••	•	•	•
	•	A	· · ·	pol	y-	-ti ole	m	<u>ی</u>	ho	alg ( t	109 1	hil	the	n h	و (	fo	r te	d	[ra	ve Hh	lir e	g e	<u>(</u> 	ר - איז	er Le	ma	n of	F	, 1 	61	lm Un Si	27	w de	onl	d 0.	2	20	ve	· · ·	•	•	• • • •
· ·	•	br	ί	r hi	λn	t	r	nì	nd	L	•	00	ų	· · ·	ſſ	VQ.V	ч (		dt	24	nd	LS		· · ·	· · · · · · · · · · · · · · · · · · ·	•	· <b>\</b> ·	•	· · ·		•	•	· · ·	•	•		•	•	· ·	•	•	•
• •	•	St	Υ0γ	g	•	ev	rid	In	ce	· ·	of	•	h	n No	Ine	22	· ·		•	•	· ·	•	•	· ·	•	•	• •	•	· ·	•	•	•	· ·	•	•	• •	•	•	• •	•	•	•
• •	•	•	• •	U	•	• •	•	•	•	•••	•	•	•	••••	•	•	••••		•	•	• •	•	•	• •	•	•	• •	•	••••	•	•	•	• •	•	•	•••	•	•	· ·	•	•	
•••	•	•	• •	•	•	• •	•	•	•	•••	•	•	•	••••	•	•	• •		•	•	• •	•	•	• •	•	0	• •	•	• •	•	•	•	• •	•	•	• •	•	•	· ·	•	•	•
	•	0	• •	•	•	• •	•	•	•	• •	•	•	•	• •	•	•	• •		•	•	••••	•	•	••••	•	0	• •	•	• •	•	•	•	• •	•	•	• •	•	0	• •	•	0	•
	•	•	••••	•	•	• •	•	0	•	• •	•	•	•	• •	•	•	• •		•	•	• •		•	• •		0	o o	•	• •	•	0		• •	•	•	• •	•	0	• •		•	

HARD PROBLEM
Weak evidence of hardness
A poly-time algorithm for Traveling Salesman Problem would solve
a problem that has resisted the efforts of thousands of
brilliant minde over many decades.
Strong evidence of hardness
A poly-time algorithm for Traveling Salesman Problem would solve
thousands of problems that have resisted the efforts of tens of
thousands of brilliant minds over many decades.

											• •																															1			
										•		•										5			•						-										•				
											ŀ	٢ı	11	<b> </b> . 1	T	Ir	J (	<u>.</u>	A		C	A	2	F.		h	11.	T I	H	(	RI	E1	D١	) (	T	0	N	L							
											. 4		/ <b>.</b>		2			۲.	<b>/</b> , <b>v</b>			<b>*</b> '		<u> </u>			· • .	1	•																
•				•			•	•		•	• •		•			•		•	•		0	•	•		•				•		•					•	• •								
				•		•	•	•	•	•	• •		•		•	•	• •	•	•			•	•		•	• •			•		•					•	• •						• •		
				•		•	•			•	• •				•	•	• •	•				•	•		•	• •				•	•					•	• •				•		• •		
•															•		• •							•	•	• •					•						• •						• •		
•					•			•							•	•	• •								•						•					•							• •		
•							•			•	• •				•	•							•		•						•					•	• •					•	• •		
															•	•								•	•						•						• •					•	• •		
															•																														
									•		• •																																		
				•		•	•	•		•	• •				•	•						•	•		•						•					•	• •								
				•		•	•	•	•	•	• •		•		•	•	• •	•	•			•	•		•	• •			•		•					•	• •						• •		1
										•	• •						• •								•	• •					•						• •						• •		
•															•		• •							•	•	• •					•						• •						• •		
•					•			•							•	•	• •														•					•						•	• •		1
•	•			•			•		•		• •				•	•				•			•	•	•	• •					•					•	• •		•	•		•	• •		
•		•		•				•			• •				•	•	•								•						•					•	• •					•	• •		
											• •																																		

•	•	•	•	•	•	•	•	•	•	•	•	•	B	50 1		Ļ	D	) <b>[</b> ]	N	G	•		4	· ·		4 4 1	S E			ہ ا بر	Т	H	 	R	E	Þ	U V V	C1	<b>[]</b>	0	<b>ا</b> ا	S	•	•	•	•	• •		•	•	•
•		¥	•	c C	ho	200	e Re		r X	e Q	ŝ	g	e	•	cl	<u>م</u>	2			•	Of	<u>)</u>	(	י יסרַ	m	Þ	n <sup>t</sup>	- A	Hiv	ې ۲	r al	•	þ		61	e er	nl	· ·	•	•	•	•	•	•	•	•	· ·		•	•	•
•	•	•	•	•	•	•	•	•	•	•		. U	•	•	0	•	•	•	•	•		• •		• •	•	l .	•	•	•	•	•	•	. <b>I</b> . 		•	•	•	• •		•	•	•	•	•	•				•	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	· ·		•	•	•	•	•	•	•	••••	•	•	•	•	· ·	•	•	•	•	•	•	•	•	• •	•		•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	· ·	•	•	•	•	•	•	•	•	• •	•	•	•	•		•	•	•	•	•	•	•	•			•	•	•
•	•	•	0		•	0	•		0	0	0	•	•	0	0	0	0	•	•	•	•	• •	0 1	• •	0	•	•	0	•	•		•	• •		•	0				•		•	•	•	•			-	•	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	••••	•	•	•	•	•	•	•	•	• •	•	•	•	•	• •	•	•	•	•	•	•	•	•	• •		•	•	•
•	•		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	• •			•	•		•		•	• •	•	0	•		• •	•	•		•	•	0		•		0	•		•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	· ·	•	•	•	•	•	•		•	• •	•	•	•		· ·	•	•	•	•	•	•	•	•	• •		•	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	• •	• •	•	•	•	•	•	•	•	•	••••	•	•	•	•	• •	•	•	•	•	•	•	•	•			•	•	•
•				0		•	•		•	•	•	•		•		•	•	•		•	•	•		• •	•			•	0	•	•	•	• •	•	•	•	•		•			•		•	0	•	• •		•	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	• •	•	•	•	•	•	•	•	•	• •	•	•	•	•	• •		•	•	•	•	•	•	•	• •		•	•	•

· · · ·	· · · · ·	· · · · · ·	BUILT	DING	A CASE	WIT	h Red	VCTION		· · · · · · ·	
X	Choos	e a la	nge class	C of	com put	ational	problem		<ul> <li>.</li> <li>.&lt;</li></ul>	.	
*	Show	that	solving	Yowr	problem	(e.g., 7	Irin (92	l solve	all	problems i	n C.
· · · ·	· · · · · ·	· · · · · · · ·		· · · · · · ·		· · · · · ·	  	· · · · · · · · · · · · · · · · · · ·	· · · ·	· · · · · · · ·	
· · · ·		·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·         · <t< td=""><td>· · · · · · · ·</td><td> </td><td>·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       &lt;</td><td>· · · · · ·</td><td> </td><td> </td><td>· · · ·</td><td> </td><td></td></t<>	· · · · · · · ·		·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·         ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       <	· · · · · ·			· · · ·		

· · · · · · · · · · · · · · · · · · ·	BUILDI	ing /	A CASE	WIT	HR	LEDU	CTIO	NS NS	· · · · ·	· · ·	
* Choose a lang	je class	C of	comput	ational	þro	blems	· · · · ·		<ul> <li></li></ul>	· · · · ·	
* Show that	solving	Yowr	problem	(eg., 7	rsp)	hil	Solve	all.	problem	<b>i</b> n	С.
* Therefore,	your	problem	n must	also	be	ont	of	nead	۰۰۰۰ ۲	· · · · · · · · · · · · · · · · · · ·	

BUILDING A CASE WITH REDUCTIONS	 	• •
* Choose a large class C of computational problems	· · · · · · · · ·	
NP-had	· · · · · · · · ·	
* Show that solving your problem (e.g., TSP) will solve all	problems in (	С.
Reduction	· · · · · · · · · · · · · · · · · · ·	· · ·
* Therefore, your problem must also be out of nearly	·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·       ·	· · · · · · · · · · · · · · · · · · ·

• •	•		• •	•	• •	•	• •	B	50	L	D		16	Î		4	C	A	SE		W	17	۲H	ļ	R	E	ÞV	C	[10	) 	S		•					
	X	۲ ۲	hor	92C	0		la	ge		cla	22	1		0	f	(	יסל י	m	þv	ta-	۲v	no	L	þ	mol	611	em	C		•		•	•			•		
· ·	¥	S	างพ	-	the	at		S	v	in (	<b>7</b>	)     	fo	W V	L	þ	roļ	61	i im		e.g	-,	T:	s P	)	4	ill		solv	re	- Q-	U	þ	nb	lms	ir ir		) 
• •	*	Th		fa	16	) )	)   	101	۲ ۲	· ·	þ	rok	10	m	• •	m	rst	 / 1		lso	• •	60			ont	<u> </u>	of	) -	ne	e Gu	h.	•	•	• •	· ·	•	• •	) n 3 n
	•	• •		•	• •	0	• •		•	• •	•	0	• •	•	•	• •	0	•	•••	•			0	• •		• •	0	•		•	• •		0		• •	•		
• •						0	• •			• •			• •		0	• •									0	• •					• •			• •				
						•								•	•													•										
					• •					• •															٠													
• •		• •			• •		• •			• •		•	• •			• •			• •					• •	٠	• •					• •	•	•		• •	•	•	
• •	•	• •		•	• •		• •			• •		•	• •		•	• •	•	•	• •	•			•	• •		• •	•	•		•	• •	•	•			•	•	
• •		• •			• •		• •	•		• •			• •			• •			• •	•	•	•		• •	٠	• •					• •	•	•	• •			•	
• •		• •			• •		• •			• •		•			•	• •		•	• •		•			• •		• •		•			• •		•	• •	• •	•	•	•
					• •				•																			•		•								

· · · · · · ·	· · · · · · ·	BUILD	ING /	A CA	ASE WIT	h Red	VCTION	S	· · · · · · · · ·	•
* Ch	oose a la	ge class	C of	Com	putational	- problem	ml	· · · · ·	· · · · · · · · ·	•
* Sho	w that	solving	yowr	prob	Im (eg., -	TSP) hri	ll solve	all p	roblands in C	•
* The	refore,	your p	roblem	must	also be	ont e	of near	h	· · · · · · · · ·	•
							· · · · · · ·		· · · · · · · · ·	•
If	any one	broblem	in C	i i i i i	broven to	be no	t boly-	time	solvable,	0
If	any one	problem	in C	ÌS	proven to	be no	t þoly-	time	solvable,	•
If they	any one so is	problem TSP.	in C	ÌS	phoven to	be no	t þoly-	time	solvable,	•
If they	any one so is	problem TSP.	in C	ÌS (	phoven to	be no	t þoly-	time	solvable,	•
If thew	any one so is	problem TSP.	in C	<u>)</u> S	proven-to	be no	t þoly-	time	solvable,	
If thew	any one so is	þroblem TSP.	in C	ĵs (	proven-to	be no	t þoly-	time	solvable,	
If they	any one so is	problem TSP.	in C	<u>)</u> S (	proven-to	be no	t þoly-	time	Solvable,	
If thew	any one so is	þroblem TSP.	in C	ĵs (	proven-to	be no	t þoly-	time	solvable,	
If they	any one So is	problem TSP.	in C	ĵs (	proven-to	be no	t þoly-	time	Solvable,	

BUILDING A CASE WITH REDUCTIONS
* Choose a large class C of computational problems
* Show that solving your problem (eg., TSP) will solve all problems in C.
* Therefore, your problem must also be out of nearh.
If any one problem in C is proven to be not poly-time solvable,
then so is TSP.
$\rho$ $h$ $\gamma$ $h$ $\gamma$ $h$ $h$ $h$ $h$
Bigger the set C =7 stronger evidence for ISPs intractability