

## **Abstract**

Word Blanks is a game in which one player has as input a secret paragraph with some words replaced with blanks with part-of-speech tags, such as plural noun, and the other player has as input the experience of being human for long enough to understand parts-of-speech. The first player queries the other player with part-of-speech tags and receives words to put in the corresponding blanks. The goal of Word Blanks is to maximize the humor of the resulting paragraph.

First, we collected data from a representative population of 11 theoretical computer science grad students as the second player in the game, then chose 5 filled paragraphs and asked 20 computer science grad students which of a pair was funnier given the order they were presented.

Second, we took our data and used a logistic regression model to learn which paragraph in any ordered pair is more humorous, and applied the model to the other filled paragraphs.

Our results are that humor dominance is a strict partial order, and that the second paragraph presented is usually the most humorous

#### 1. Introduction

Formally, Word Blanks is a family of functions  $\mathcal{WB} = \{\mathbf{wb}_k\}_{k \in \mathbb{N}}$  each taking a secret paragraph p and an response function r and returning a filled paragraph in the human experience monad  $\mathbb{H}$ .

$$\begin{split} p &:: \P = \mathsf{PoS}^k \times (\mathsf{W}^k \to \mathbb{H} \, \mathsf{W}^n), k \leq n \\ r &:: \Re = \mathsf{PoS}^k \to \mathbb{H} \, \mathsf{W}^k \\ \mathsf{wb}_k &:: \P \to \Re \to \mathbb{H} \, \mathsf{W}^n \\ \mathsf{wb}_k &p r = r \, p_0 >> = p_1 \end{split}$$

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, contact the Owner/Author. Request permissions from permissions@acm.org or Publications Dept., ACM, Inc., fax +1 (212) 869-0481. Copyright held by Owner/Author. Publication Rights Licensed to ACM.

Copyright © ACM [to be supplied]...\$15.00 DOI: http://dx.doi.org/10.1145/(to come)

Tag	Part of Speech	Your Word
JJ	ordinal adjective or numeral	
NN	noun	
NNS	plural noun	
JJ	ordinal adjective or numeral	
NNP	singular proper noun	
NNS	plural noun	
VBN	past participle verb	
NNP	singular proper noun	
VBD	past tense verb	
NNS	plural noun	
NN	noun	
VBN	past participle verb	
JJ	ordinal adjective or numeral	
VBZ	3rd p. sing. present tense verb	

**Table 1.** Input format

We investigated the Word Blanks humor relation  $\hbar$ , which takes two filled-in paragraphs and determines which one is funnier in  $\mathbb{H}$ .

$$\hbar :: \mathsf{W}^n \to \mathsf{W}^n \to \mathbb{H} \ 2$$

## 2. The Text

Given our audience, we secretly chose our paragraph as the abstract of a notable theoretical computer science paper.

## 2.1 Original Quote

"A large class of computational problems involve the determination of properties of graphs, digraphs, integers, arrays of integers, finite families of finite sets, boolean formulas and elements of other countable domains. Through simple encodings from such domains into the set of words over a finite alphabet these problems can be converted into language recognition problems, and we can inquire into their computational complexity. It is reasonable to consider such a problem satisfactorily solved when an algorithm for its solution is found which terminates within a number of steps bounded by a polynomial in the length of the input. We show that a large number of classic unsolved problems of covering, matching, packing, routing, assignment and sequencing are equivalent, in the sense that either each of them possesses a polynomial-bounded algorithm or none of them does." [4]

## 2.2 The Word Blanks Game

We used nltk, Python's Natural Language Tool Kit [3], to generate part-of-speech tags. Due to its mistakes, the paragraph was modified slightly as follows:

The class of c	computational problei
ordinal adjective or numeral	
involve the properties	of graphs, digraphs,
noun	
integers, of integers, finite fan	nilies of
plural noun	
sets,	-ean formulas
ordinal adjective or numeral singular prope	er noun
and elements of other countable domain	ins. Through simple
	0 1
encodings from such domains into the se	et of
over a finite alphabet these	a probleme can be
	e problems can be
plural noun into	racamitian problems
past participle verb into singular proper noun r	ecogiiition problems,
and we can inquire into their compu	stational complexity
and we can inquire into their compo	itational complexity.
It is reasonable to consider such a p	roblem satisfactorily
when an algorithm for it	s solution is found
past tense verb	s solution is lound
which terminates within a number of	bounded
	nlural noun
by a polynomial in the of the	he input We show
that a large number of classic past pa	problems
nact na	rticiple verb
of covering, matching, packing, routing	no assionment and
or covering, matering, packing, routin	ng, ussignment und
sequencing are	-order equivalent, in
ordinal adjective or numeral	1,
the sense that either each of them	
	. sing. present tense verb
a polynomial - bounded algorithm or no	
- ry commerc angerrann or no	

# 3. Procedure

In the first part of the experiment, data sources were recruited at Theory Tea and asked to fill in copies of Table 2.1 printed on slips of paper. The sources were not told what their input would be used for, and we replaced their names with NP-Complete problems for anonymity. 11 sets of words were collected in total.

Three other sets of words were also used, which were generated as follows:

- One set was randomly generated using the lists made available at [1].
- One set was generated using the next word prediction features of the Messaging app Version 4.4.2-G730AUCUBNG4 on an Android cellular phone.
- One set was the actual set of words used in [4].

The sets of words thusly obtained were used to complete the Word Blanks game from Section 2.2 to generate test paragraphs. Out of these, a set of 5 test paragraphs were selected according to the whims of the authors.

In the next part of the experiment, a random set of 20 evaluators were chosen based on their proximity to application. After an apology for the rude interruption from their research, each was presented with two test paragraphs, one at a time, and asked to read them without being told what for. After the evaluator had read both

paragraphs, they were asked which of the two they found funnier. For each evaluator, the identity of the paragrphs presented, the order in which they were presented, and the identity of the one judged funnier were recorded. From 20 evaluators, we had one comparison for each pair of paragraphs in either order of presentation.

Once this data was obtained, Logistic Regression (specifically, the implementation in the scikit-learn Python package [2]) was used to predict the results of comparisons between the remaining pairs. The input data for the regression was generated as follows:

- 1. Each character in each word was converted to its ASCII value.
- 2. For each blank, all words corresponding to that blank across all sets of words were padded with zeros to be of the same length.
- For each set, these padded ASCII words were concatenated to yield one vector of numbers.
- 4. For each pair of sets, their vectors were concatenated to yield the final input vector.

The set of evaluations obtained from the evaluators was then used as training data to the logistic regression-based classifier, which was then used to predict results of comparisons between all pairs of sets of words.

## 4. Results

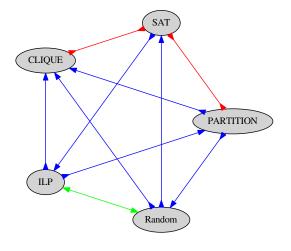


Figure 1. Graphical representation of collected evaluations

The results from the evaluators is presented in Tables 2 and 3. The test paragraphs used are included in Appendix A.1. The same is represented graphically in Figure 1. In all diagrams and tables, blue represents that one paragraph in the pair was funnier than the other whether it was presented first or second; green represents that the first paragraph presented was always funnier, and red that the second paragraph presented was always funnier. In the graphs, blue arrows have both ends pointing in the same direction (away from the funnier paragraph), green arrows point outward, and red inward.

The following interesting observations may be drawn from our experimental data:

- 75% of theorists are more funny when they go second, whereas random won most when it went first.
- ILP (a theorist) is isomorphic to Random.
- The two best performing theorists both referenced U.S. presidents or presidential candidates.

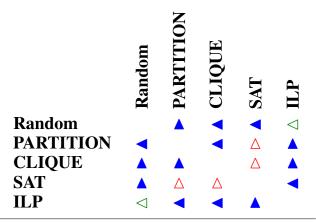


Table 2. Collected evaluations. The paragraph corresponding to the author on the left was shown first. Arrows point to the winner in each comparison.

		Random	<b>PARTITION</b>	CLIQUE	SAT	ILP
Wins as	$\mathbf{1^{st}}$	3	2	0	1	3
Wins as	2 <sup>nd</sup>	2	3	1	3	2
Wins	total	5	5	1	4	5

**Table 3.** Number of victories when presented first, second, and in total, from collected evaluations

- The two best performing theorists both used concrete numbers as their "-order equivalent" word
- 42 was put by **SAT** and **ILP** in the same blank.

The results of the predictions of the model is represented in Table 4 tabularly and Figure 3 graphically, whereas Figure 2 contains just the blue "dominance" arrows. Our striking result is that humor dominance is a strict partial order, meaning it is anti-reflexive and transitive.

When considering how varying the ordering affects the relative humor of Word Game pairs, we noticed that our model predicted 21 blue arrows, 22 green arrows, and 51 red arrows, thus that the number of second-place victories is  $\frac{123}{188} = 0.6542553191489362\%$ , compared to a mere  $\frac{65}{188} = 0.34574468085106386\%$  first-place victories. Thus, a paragraph is nearly 1.8923076923076922 times more likely to be more humorous if it appears second.

## Acknowledgements

We would like to thank our test subject theorists for volunteering their human experience to science: G, Will, Nick, Adam, Aviv, Sam, Shalev, Pritish, Rio, Prashant, and Atalay.

# References

- [1] Randomlists.com. http://www.randomlists.com/. Accessed:
- [2] scikit-learn. http://scikit-learn.org/. Accessed: 2016-02-

- [3] S. Bird, E. Klein, and E. Loper. Natural Language Processing with Python. O'Reilly Media, 2009.
- [4] R. M. Karp. Reducibility among combinatorial problems. In Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York., pages 85-103, 1972. URL http: //www.cs.berkeley.edu/~luca/cs172/karp.pdf.

## **Paragraphs**

# **Test Paragraphs**

Th

The	ese w	ere the pa	aragra	phs actua	ally p	resent	ted to	evalu	ators	•	
1.	Ran	dom:									
	The	eigt	h	class o	of cor	nputa	tional	prob	lems	inv	olve
	the	paper		propertie	es of	grap	hs, di	grap	hs, i	nteg	gers,
	1	ables	of in	tegers, fi	nite f	amilie	s of [	tv	wo		sets,
	F	Freddie M	ercur	y -ean	forn	nulas	and e	leme	nts o	of c	ther
		table dor				<u>.                                    </u>		_			
	mair	is into the	set c	of nos	ses	ove	r a fin	ite al	phab	et t	hese
	prob	lems can	be	tickled	d	into	Rob	ert D	own	ey J	r.
		gnition prole									
	satis	factorily	ro	bbed	when	an a	gorith	m fo	r its	solu	ıtion
	is fo	und which	h ter	minates v	withii	ı a nı	ımber	of [	are	ches	S
	bour	ided by a	polyn	omial in	the	cro	wd	of t	he in	put	. We
		that a la	_				ubtrac ssignn		prob and s		
	ing a	are 1	ast	-order	equiv	alent	in the	sen	se tha	at ei	ither
	each	of them	j	umps	a po	olynor	nial-bo	ound	ed al	gori	ithm
	or no	one of the	m do.		1						
2.	PAR	TITION	:								
	The	doub	ole	class c	of cor	nputa	tional	prob	lems	inv	olve
	the	bathroo		propertie			_				-
		-4	L c :	c.	.: c	:1: -		_:	4		4_

#### 2

THE	dout	16	Class 0	COIII	puta	попа	prot	леш	SIIIV	OIVE
the	bathroc	m j	propertie	s of	grap	hs, c	ligrap	ohs,	inte	gers,
	stars	of inte	egers, fir	nite fa	milie	es of [	si	xteer	ı	sets,
	Barac		- 1	form						
coun	table don	nains.	Through	simp	le ei	ncodi	ngs f	rom	such	ı do-
main	s into the	set of	sho	es	ove	er a fi	nite a	lpha	bet 1	these
	lems can		eaten					ı Goı		
recog	gnition pr	oblem	s, and w	e can	inq	uire i	nto tł	neir o	com	puta-
tiona	d comple	xity. It	is reaso	nable	to c	onsid	er su	ch a	pro	blem
satis	factorily [	bar	fed	when	an a	lgorit	hm fo	or its	sol	ution
is fo	und whic	h tern	ninates v	vithin	a nı	umbei	r of [	sv	vamj	ps
boun	ided by a	polyno	omial in	the	but	tox	of	the i	nput	t. We
show	that a la	rge nu	mber of	classi	c	holp	en	pro	bler	ns of
cove	ring, mate	ching,	packing,	routi	ng, a	ssign	ment	and	sequ	ienc-
ing a	re 32,42	26,941	-order	equiva	lent	, in th	ie ser	ise tl	nat e	ither
each	of them	bo	eats	a pol	ynoı	nial-t	ounc	led a	lgoi	rithm
or no	one of the	m do.								

#### 3. CLIQUE:

The	ntł	ı	class of	al problem	s involve		
the	wolveri	ne	properties	of	graphs,	digraphs,	integers
wo	lverines	of in	tegers, finit	e fa	milies of	i	sets
			-ean f	orm	ulas and	elements	of other

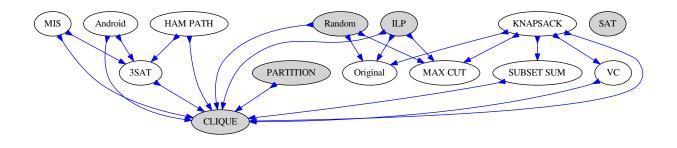


Figure 2. Graphical representation predicted complete dominations.

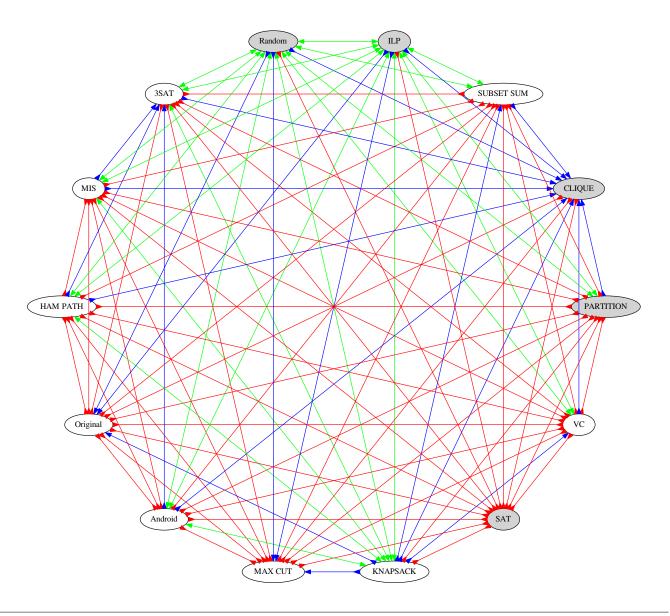


Figure 3. Graphical representation of all predictions.

	VC	PARTITION	KNAPSACK	MAX CUT	Android	Original	HAM PATH	MIS	3SAT	Random	ILP	SUBSET SUM	CLIQUE	SAT
VC		Δ	<b>^</b>	Δ	Δ	Δ	Δ	Δ	Δ	$\Diamond$	$\triangleleft$	Δ	•	Δ
PARTITION	Δ		Δ	Δ	Δ	Δ	Δ	Δ	Δ	$\triangleleft$	(⊲)	Δ	•	Δ
KNAPSACK	•	Δ		•	$\triangleleft$	•	$\Diamond$	$\triangleleft$	$\triangleleft$	$\triangleleft$	$\triangleleft$	•	•	Δ
MAX CUT	Δ	Δ	<b>A</b>		Δ	Δ	Δ	Δ	Δ	<b>A</b>	<b>A</b>	Δ	Δ	Δ
Android	Δ	Δ	$\triangleleft$	Δ		Δ	Δ	Δ	•	$\triangleleft$	$\triangleleft$	Δ	•	Δ
Original	Δ	Δ	<b>A</b>	Δ	Δ		Δ	Δ	Δ	<b>A</b>	<b>A</b>	Δ	Δ	Δ
HAM PATH	Δ	Δ	$\triangleleft$	Δ	Δ	Δ		Δ	•	$\triangleleft$	◁	Δ	•	Δ
MIS	Δ	Δ	$\triangleleft$	Δ	Δ	Δ	Δ		•	$\triangleleft$	$\triangleleft$	Δ	•	Δ
3SAT	Δ	Δ	$\triangleleft$	Δ	<b>A</b>	Δ	<b>A</b>	<b>A</b>		$\triangleleft$	◁	Δ	•	Δ
Random	$\triangleleft$	(⊲)	$\triangleleft$	•	$\triangleleft$	•	◁	$\triangleleft$	$\triangleleft$		◁	$\triangleleft$	•	( <u>\( \( \( \) \) \)</u>
ILP	◁	◁	$\triangleleft$	•	$\triangleleft$	•	◁	$\triangleleft$	$\triangleleft$	$\triangleleft$		$\triangleleft$	•	Δ
SUBSET SUM	Δ	Δ	<b>A</b>	Δ	Δ	Δ	Δ	Δ	Δ	$\triangleleft$	$\triangleleft$		•	Δ
CLIQUE	<b>A</b>	<b>A</b>	<b>A</b>	Δ	<b>A</b>	Δ	<b>A</b>	<b>A</b>	<b>A</b>	<b>A</b>	<b>A</b>	<b>A</b>		Δ
SAT	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	( <u>\( \( \( \) \) </u>	Δ	Δ	

**Table 4.** Model predictions for comparisons between all test paragraphs. The training people are highlighted with grey backgrounds. The paranthesised entries differ from the training data.

mains proble recogn tional satisfa is four	able domains. Through simple encodings from such do- into the set of cabinets over a finite alphabet these ems can be capped into into mitton problems, and we can inquire into their computa- complexity. It is reasonable to consider such a problem actorily sat when an algorithm for its solution and which terminates within a number of parakeets and by a polynomial in the reefer of the input. We	bounded by a polynomial in the spoon of the input. We show that a large number of classic missed problems of covering, matching, packing, routing, assignment and sequencing are minus one order equivalent, in the sense that either each of them sleeps a polynomial-bounded algorithm or none of them do.  5. ILP (Integer Linear Programming):
show to coveri ing are	that a large number of classic smoked problems of ng, matching, packing, routing, assignment and sequenc-	The 82 class of computational problems involve the orangutan properties of graphs, digraphs, integers kids of integers, finite families of 42 sets  Donald ean formulas and elements of other countable domains. Through simple encodings from such domains into the set of monkeys over a finite alphabet these
counta mains proble recogr tional satisfa	zeroth class of computational problems involve pebble properties of graphs, digraphs, integers, wers of integers, finite families of 42 sets,  Property ean formulas and elements of other able domains. Through simple encodings from such dointo the set of capes over a finite alphabet these ems can be lost into Anarchy mition problems, and we can inquire into their computation problems, and we can inquire into their computation problems, and when an algorithm for its solution and which terminates within a number of canes	problems can be blown into Bernie recognition problems, and we can inquire into their computational complexity. It is reasonable to consider such a problem satisfactorily trumped when an algorithm for its solution is found which terminates within a number of ducks bounded by a polynomial in the wand of the input. We show that a large number of classic danced problems of covering, matching, packing, routing, assignment and sequencing are 163 order equivalent, in the sense that either each of them sings a polynomial-bounded algorithm or none of them do.

# A.2 Other Paragraphs

These were the paragraphs that we only predicted the outcomes of.

1	VC	(Vertex	Cover	١
1.	V (	I VELLEX	COVEL	,

The	fift	h	class c	of com	puta	tiona	l prot	olems	s inv	volve
the	tea	]	properti	es of	grap	hs, o	digrap	ohs,	inte	gers,
ŀ	ooxes	of inte	egers, fi	nite fa	milie	es of	thi	rteer	ı	sets,
	Grino	h	-ean	form	ulas	and	elem	ents	of	other
coun	table dor	nains.	Through	h simp	ole ei	ncodi	ngs f	rom	sucl	n do-
main	s into the	e set of	comp	outers	ove	er a fi	nite a	lpha	bet	these
prob	lems can	be	brough	nt i	nto					
recos	gnition pi	oblem	s, and v	ve can	inai	uire i	nto tł	neir c	om	puta-
	l comple									
	factorily			when						
	٠ ١					_	-			
is fo	und whic	ch tern	nnates	within	a nı	ımbe	r of	p	ants	3
boun	ded by a	polyno	omial in	the	ban	ana	of	the i	npu	t. We
show	that a la	rge nu	mber of	classi	с	don	ne	pro	bler	ns of
cove	ring, mat	ching,	<u>p</u> acking	, routi	ng, a	ssign	ment	and	sequ	ienc-
ing a	ire sev	enth	-order	equiva	ılent.	, in tł	ne ser	ise th	ıat e	either
each	of them	5	sits	a pol	ynor	nial-l	ounc	led a	lgoı	rithm
or no	one of the	m do.		•						

## 2. MAX CUT:

The	eleventy	-first	class o	of co	mput	ational	prob	olem	s in	volve
the	justice	p	roperti	es o	f gra	phs, d	igrap	hs,	inte	gers,
k	ittens	of inte	gers, fi	nite 1	famil	es of	Z	zero		sets,
William ean formulas and elements of other countable domains. Through simple encodings from such do-										
mains into the set of Canadians over a finite alphabet these										
	problems can be invented into Socrates									
tiona	recognition problems, and we can inquire into their computational complexity. It is reasonable to consider such a problem									
satis	factorily	interce	epted	whe	n an	algorith	ım fo	or its	sol	ution
is fo	und whic	h term	inates	withi	n a i	number	of [	d	onu	is
bour	ided by a	polyno	mial in	the	tri	angle	of	the i	npu	t. We
show	that a la	rge nun	nber of	clas	sic	sougl	nt	pro	bler	ns of
cove	covering, matching, packing, routing, assignment and sequenc-									
ing are minus fifth order equivalent, in the sense that either										
each	of them	dest	roys	a p	olyno	mial-b	ound	led a	algoi	rithm
or no	or none of them do.									

# 3. MIS (Independent Set):

The	nını	h	class of computational problems involv							
the	geome	try	properties of graphs, digraphs, intege							
	dogs	of in	tegers, finite	famili	es of	three	sets,			
	me -ean formulas and elements of other									
cour	ntable dor	nains.	Through sin	nple e	ncodi	ngs from	such do-			
mair	ns into the	e set c	of burgers	ove	er a fi	nite alpha	bet these			
	lems can		killed	into		you				
reco	gnition p	robler	ns, and we ca	an inq	uire i	nto their	computa-			
tiona	al comple	xity. 1	It is reasonab	le to c	consid	der such a	problem			

satisfact	orily	dea	d	when a	n algorithi	n fo	r its solution		
is found which terminates within a number of fries									
bounded by a polynomial in the chess of the input. We									
show that a large number of classic pwned problems of									
covering, matching, packing, routing, assignment and sequenc-									
ing are	zei	oth	-order	equival	ent, in the	sen	se that either		
each of	them	pla	ıys	a poly	nomial-bo	und	ed algorithm		
or none of them do.									

## 4. SUBSET SUM:

The	uncoun	table	_ class c	of com	putation	al proble	ms ir	ivolve		
the	fish		properties of graphs, digraphs, inte							
5	sheep	of in	tegers, fi	nite fa	milies of	$\omega^{\omega}$	υ	sets,		
Turing -ean formulas and elements of other										
coun	countable domains. Through simple encodings from such do-									
main	mains into the set of iPhones over a finite alphabet these									
	lems can	L			nto	Babb	_			
recog	gnition pi	obler	ns, and v	ve can	inquire	into thei	r con	iputa-		
tiona	l comple	xity.	It is reas	onable	to consi	der such	a pro	oblem		
satis	factorily [	e	aten	when	an algori	thm for	its so	lution		
is fo	und whic	h ter	minates	within	a numb	er of	finge	ers		
boun	bounded by a polynomial in the chicken of the input. We									
show	that a la	rge n	umber of	classic	cal	led p	roble	ms of		
covering, matching, packing, routing, assignment and sequenc-										
ing a	ire coui	ntable	order-	equiva	lent, in	he sense	that	either		
	of them		wims	a pol	ynomial-	-bounded	d algo	rithm		
or no	one of the	m do								

## 5. KNAPSACK:

The	13t	h	class o	of con	nputa	tional p	orob	lems	s invo	olve
the	chess	s p	properties of graphs, digraphs, integers,							ers,
boobs of integers, finite families of 1 se								ets,		
	London ean formulas and elements of other countable domains. Through simple encodings from such do-									
coun	itable dor	nains.	I'hrougl	n sim	iple e	ncoding	gs fr	om	such	do-
mair	ns into the	e set of	glas	ses	ove	er a fini	te al	lpha	bet th	iese
prob	lems can	be	beater	l	into		Mo	scov	V	
reco	gnition p	roblems	s, and v	ve ca	n inq	uire int	o th	eir c	comp	uta-
	al comple									
	· 1								•	
satis	factorily	riac	ien	wner	i an a	lgorith	n 10	r its	solu	tion
is fo	ound which	h term	inates v	withi	n a n	umber	of [	ŀ	nairs	
bour	nded by a	polyno	mial in	the	ho	rse	of t	he i	nput.	We
show that a large number of classic done problems of										
covering, matching, packing, routing, assignment and sequenc-										
ing a	are 3	Brd	-order	equi	valent	, in the	sen	se th	at ei	ther
each	of them	sw	ims	a po	olyno	mial-bo	und	ed a	lgori	thm
or no	or none of them do.									

## 6. **3SAT:**

The	many		class of	class of computational problems in					
the	spoor	1 ]	properties	of	graphs,	digraphs,	integers,		
at	tempts	of int	egers, finit	e fa	milies of	3	sets.		

MIT	-ean	formu	las and el	lements	of other				
countable domains. Through simple encodings from such do-									
mains into the set of students over a finite alphabet these									
problems can be proven into Harvard									
recognition pro	recognition problems, and we can inquire into their computa-								
tional complex	ity. It is reaso	onable t	o conside	r such a	problem				
satisfactorily	falsified	when a	n algorith	m for its	solution				
is found which	terminates	within a	number	of pro	fessors				
bounded by a p	olynomial in	the	proof	of the i	nput. We				
show that a larg	ge number of	classic	struck	pro	blems of				
covering, match	ning, packing	, routing	g, assignm	ent and	sequenc-				
ing are 12	2 -order	equival	ent, in the	sense t	hat either				
each of them	tries	a poly	nomial-bo	ounded a	algorithm				
or none of then	ı do.	-							

7. <b>HA</b> !	M PATH:								
The	zero	zeroth class of computational problems involv							
the	chees	cheese properties of graphs, digraphs, integers,							
9	glasses of integers, finite families of tenth sets,								
	Sealand ean formulas and elements of other countable domains. Through simple encodings from such domains into the set of dinghies over a finite alphabet these								
reco tion:	problems can be stolen into Mona Lisa recognition problems, and we can inquire into their computational complexity. It is reasonable to consider such a problem satisfactorily jumped when an algorithm for its solution								
is fo	is found which terminates within a number of sharks								
bounded by a polynomial in the moon of the input. We									
show that a large number of classic fallen problems of covering, matching, packing, routing, assignment and sequenc-									
ing a	ing are fifth -order equivalent, in the sense that either								
	each of them kicks a polynomial-bounded algorithm or none of them do.								