

# Solutions to Fermat's Last Theorem \*

Robert Bernecky  
Snake Island Research Inc  
18 Fifth Street, Ward's Island  
Toronto, Ontario M5J 2B9  
Canada  
+1 416 203 0854  
bernecky@acm.org

## Abstract

This article describes our experience with solutions to a sticky problem in mathematics.

```
lab←'0123456789:' ⍝ Find labels
lab←lab,'Δabc...xyz'
lab←lab,'ΔABC...XYZ'
lab←v/(^fnt∈lab)^fnt=':'
```

## 1 Introduction

This is the intro.

## 2 Another section

Here is another section.

### 2.1 Here's a subsection

You can display APL code, such as plus reduce iota x assigned 9, this way: `+/ιx←9`. The slash-0 is used to delimit a macro name from its argument, when no blank is wanted between the expanded macro text and the next character.

You can put text in tables this way:

---

\*Parts of this originally appeared in the APL98 Conference Proceedings.[Ber98]

### 3 The complete APL character set at 11pt

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0		;	.													
1						┆	L	⊥	J		—	<i>TM</i>	©	®		
2		!	"	#	\$	%	&	'	(	)	*	+	,	—	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8			,	f	"	...	†	‡	^	%	S	<	œ			
9		\	/	"	"	•	—	—	~	<i>TM</i>	S	>	œ			
10	⌘	..	—		≤		≥		≠	√		×	;	⌘	∴	≠
11	⋆	;		ℓ	℥		⌈	•		⋈		⊖	⊖	≡	⌈	⌘
12		α	⊥	η	L	∈		∇	Δ	ι	◦		□		τ	○
13			ρ	┘		↓	u	ω	⊃	↑	⊂	┘	⋈	⌈		÷
14	I	Θ	⊕	⌘		⊆	∇	∇	Δ	⊥	⋈		⊖		⊖	ö
15	⊗				⊗		⋈	⊕			⊕	←	Δ	→	◇	⋈

#### 4 The complete APL character set at \huge

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0		¯	.													
1						⌈	⌊	⌋	⌌		—	<i>TM</i>	©	®		
2		!	"	#	\$	%	&	'	(	)	*	+	,	—	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8			,	f	"	...	†	‡	^	%	S	<	€			
9		\	'	"	"	•	—	—	~	<sup>TM</sup>	S	>	œ			
10	⌈	⋯	—		≤		≥		≠	√		×	,	⊠	∴	≠
11	⋈	;		ℓ	¥		¬	•	≈		⊠	⊠	≡	⊠	⋈	⋈
12		α	⊥	∩	⊂	∈		∇	Δ	ℓ	◦		□		⊤	○
13			ρ	⌈		↓	∪	ω	⊃	↑	⊂	⊢	≠	⊢		÷
14	⊥	Θ	⊕	⊞		⊆	∇	∇	⊠	⊂	⋈		⊠		⊕	⋈
15	⊗				⊗		⊗	⊕			⊕	←	⊠	→	◇	⋈

#### 5 Summary and Future Work

Next, we plan to figure out how to factor large numbers quickly.

## 6 Acknowledgements

Thanks, Mom.

## References

- [Ber98] Robert Bernecky. EGREGION: A branch coverage tool for APL. In Sergio Picchi and Marco Micocci, editors, *APL98 Conference Proceedings*, pages 1–20. APL Italiana, July 1998.