微信公众号：学习资料杂货铺

二维码

公众号创建人：一名普通理工科学生，对数学、文学、软件、电影、动漫及运动有兴趣。也会收集高中数、理、化、生的竞赛，自招，高考的资料

非专业人士，都是个人爱好及部分主观感受。爱好自由，喜欢独立思考，偶尔与主流思想背道而驰。缺少的部分会尽量陆陆续续补充

喜欢收集信息、分享资料。

欢迎大家的关注

The Comprehensive LaTeX Symbol List

Scott Pakin <pakin@uiuc.edu>*

29 September 2003

Abstract

This document lists 2826 symbols and the corresponding LaTeX commands that produce them. Some of these symbols are guaranteed to be available in every LaTeX 2_ε system; others require fonts and packages that may not accompany a given distribution and that therefore need to be installed. All of the fonts and packages used to prepare this document—as well as this document itself—are freely available from the Comprehensive TeX Archive Network (`http://www.ctan.org`).

Contents

*The original version of this document was written by David Carlisle, with several additional tables provided by Alexander Holt. See Section 7.6 on page 69 for more information about who did what.

1 Introduction

Welcome to the Comprehensive LaTeX Symbol List! This document strives to be your primary source of LaTeX symbol information: font samples, LaTeX commands, packages, usage details, caveats—everything needed to put thousands of different symbols at your disposal. All of the fonts covered herein meet the following criteria:

1. They are freely available from the Comprehensive TeX Archive Network (`http://www.ctan.org`).
2. All of their symbols have LaTeX 2_ε bindings. That is, a user should be able to access a symbol by name, not just by `\char`⟨*number*⟩.

These are not particularly limiting criteria; the Comprehensive LaTeX Symbol List contains samples of 2826 symbols—quite a large number. Some of these symbols are guaranteed to be available in every LaTeX 2_ε system; others require fonts and packages that may not accompany a given distribution and that therefore need to be installed. See `http://www.tex.ac.uk/cgi-bin/texfaq2html?label=instpackages+wherefiles` for help with installing new fonts and packages.

1.1 Document Usage

Each section of this document contains a number of font tables. Each table shows a set of symbols, with the corresponding LaTeX command to the right of each symbol. A table's caption indicates what package needs to be loaded in order to access that table's symbols. For example, the symbols in Table 23, "textcomp Old-Style Numerals", are made available by putting "`\usepackage{textcomp}`" in your document's preamble. "$\mathcal{AMS}$" means to use the $\mathcal{AMS}$ packages, viz. amssymb and/or amsmath. Notes below a table provide additional information about some or all the symbols in that table.

One note that appears a few times in this document, particularly in Section 2, indicates that certain symbols do not exist in the OT1 font encoding (Donald Knuth's original, 7-bit font encoding, which is the default font encoding for LaTeX) and that you should use fontenc to select a different encoding, such as T1 (a common 8-bit font encoding). That means that you should put "`\usepackage[`⟨*encoding*⟩`]{fontenc}`" in your document's preamble, where ⟨*encoding*⟩ is, e.g., `T1` or `LY1`. To limit the change in font encoding to the current group, use "`\fontencoding{`⟨*encoding*⟩`}\selectfont`".

Section 7 contains some additional information about the symbols in this document. It shows which symbol names are not unique across packages, gives examples of how to create new symbols out of existing symbols, explains how symbols are spaced in math mode, presents a LaTeX ASCII and Latin 1 tables, and provides some information about this document itself. The Comprehensive LaTeX Symbol List ends with an index of all the symbols in the document and various additional useful terms.

1.2 Frequently Requested Symbols

There are a number of symbols that are requested over and over again on `comp.text.tex`. If you're looking for such a symbol the following list will help you find it quickly.

2 Body-text symbols

This section lists symbols that are intended for use in running text, such as punctuation marks, accents, ligatures, and currency symbols.

Table 1: LaTeX 2_ε Escapable "Special" Characters

$	\$	%	\%	_	_*	}	\}	&	\&	#	\#	{	\{

* The underscore package redefines "_" to produce an underscore in text mode (i.e., it makes it unnecessary to escape the underscore character).

Table 2: LaTeX 2_ε Commands Defined to Work in Both Math and Text Mode

$	\$	_	_	‡	\ddag	{	\{
¶	\P	©	©	\copyright	…	\dots	}	\}
§	\S	†	\dag	£	\pounds		

Where two symbols are present, the left one is the "faked" symbol that LaTeX 2_ε provides by default, and the right one is the "true" symbol that textcomp makes available.

Table 3: Predefined LaTeX 2_ε Text-mode Commands

	^	\textasciicircum		<	\textless
	~	\textasciitilde	a	a	\textordfeminine
	*	\textasteriskcentered	o	o	\textordmasculine
	\	\textbackslash		¶	\textparagraph
	|	\textbar		·	\textperiodcentered
	{	\textbraceleft		¿	\textquestiondown
	}	\textbraceright		“	\textquotedblleft
	•	\textbullet		”	\textquotedblright
©	©	\textcopyright		‘	\textquoteleft
	†	\textdagger		’	\textquoteright
	‡	\textdaggerdbl	®	®	\textregistered
	$	\textdollar		§	\textsection
	…	\textellipsis		£	\textsterling
	—	\textemdash	TM	TM	\texttrademark
	–	\textendash		_	\textunderscore
	¡	\textexclamdown		␣	\textvisiblespace
	>	\textgreater			

Where two symbols are present, the left one is the "faked" symbol that LaTeX 2_ε provides by default, and the right one is the "true" symbol that textcomp makes available.

Table 4: Non-ASCII Letters (Excluding Accented Letters)

å	\aa	Đ	\DH*	Ł	\L	ø	\o	ß	\ss
Å	\AA	ð	\dh*	ł	\l	Ø	\O	SS	\SS
Æ	\AE	Đ	\DJ*	Ŋ	\NG*	Œ	\OE	Þ	\TH*
æ	\ae	đ	\dj*	ŋ	\ng*	œ	\oe	þ	\th*

* Not available in the OT1 font encoding. Use the fontenc package to select an alternate font encoding, such as T1.

Table 5: Letters Used to Typeset African Languages

Ð	\B{D}	ƈ	\m{c}	ƒ	\m{f}	ƙ	\m{k}	ţ	\M{t}	ȝ	\m{Z}
đ	\B{d}	Ɗ	\m{D}	Ƒ	\m{F}	Ɲ	\m{N}	Ƭ	\M{T}	Ɛ̃	\T{E}
Ħ	\B{H}	ɖ	\M{d}	Ɣ	\m{G}	ŋ	\m{n}	ƭ	\m{t}	ɛ̃	\T{e}
ħ	\B{h}	Ð	\M{D}	ɣ	\m{g}	ɔ	\m{o}	Ʈ	\m{T}	Õ	\T{O}
ŧ	\B{t}	ƈ	\m{d}	Ɩ	\m{I}	Ɔ	\m{O}	ʊ	\m{u}*	õ	\T{o}
Ŧ	\B{T}	Ɛ	\m{E}	ɩ	\m{i}	Ƥ	\m{P}	Ʊ	\m{U}*		
ɓ	\m{b}	ɛ	\m{e}	Ɲ	\m{J}	ƥ	\m{p}	Ƴ	\m{Y}		
Ɓ	\m{B}	Ǝ	\M{E}	ɲ	\m{j}	ʃ	\m{s}	ƴ	\m{y}		
Ƈ	\m{C}	ə	\M{e}	Ƙ	\m{K}	Ʃ	\m{S}	ʒ	\m{z}		

These characters all need the T4 font encoding, which is provided by the fc package.

* \m{v} and \m{V} are synonyms for \m{u} and \m{U}.

Table 6: Punctuation Marks Not Found in OT1

«	\guillemotleft	‹	\guilsinglleft	„	\quotedblbase	"	\textquotedbl
»	\guillemotright	›	\guilsinglright	‚	\quotesinglbase		

To get these symbols, use the fontenc package to select an alternate font encoding, such as T1.

Table 7: pifont Decorative Punctuation Marks

❛	\ding{123}	❝	\ding{125}	❡	\ding{161}	❣	\ding{163}
❜	\ding{124}	❞	\ding{126}	❢	\ding{162}		

Table 8: wasysym Phonetic Symbols

Ð	\DH	ð	\dh	ɔ	\openo
Þ	\Thorn	ə	\inve	þ	\thorn

Table 9: tipa Phonetic Symbols

ɤ	\textbabygamma	ʔ	\textglotstop	ɳ	\textrtailn
ƀ	\textbarb	ʻ	\texthalflength	ɽ	\textrtailr
ȼ	\textbarc	ъ	\texthardsign	ʂ	\textrtails
đ	\textbard	ʰ	\texthooktop	ʈ	\textrtailt
ʄ	\textbardotlessj	ɓ	\texthtb	ʐ	\textrtailz
ǥ	\textbarg	ʄ	\texthtbardotlessj	˞	\textrthook
ʡ	\textbarglotstop	ƈ	\texthtc	ᴀ	\textsca
ɨ	\textbari	ɗ	\texthtd	ʙ	\textscb
ɬ	\textbarl	ɠ	\texthtg	ᴇ	\textsce
ɵ	\textbaro	ɦ	\texthth	ɢ	\textscg
ʢ	\textbarrevglotstop	ʄ	\texththeng	ʜ	\textsch
ʉ	\textbaru	ƙ	\texthtk	ə	\textschwa
ɮ	\textbeltl	ɸ	\texthtp	ɪ	\textsci
β	\textbeta	ʠ	\texthtq	ᴊ	\textscj
ʘ	\textbullseye	ʠ	\texthtrtaild	ʟ	\textscl
ˊ	\textceltpal	ƈ	\texthtscg	ɴ	\textscn
χ	\textchi	ƫ	\texthtt	œ	\textscoelig
ɔ	\textcloseepsilon	ʰʊ	\texthvlig	ɷ	\textscomega
ɷ	\textcloseomega	ʖ	\textinvglotstop	ʀ	\textscr
ɜ	\textcloserevepsilon	ʁ	\textinvscr	ɑ	\textscripta
ʒ	\textcommatailz	ɩ	\textiota	ɡ	\textscriptg
˥	\textcorner	ʎ	\textlambda	ʊ	\textscriptv
ƀ	\textcrb	ː	\textlengthmark	ᴜ	\textscu
đ	\textcrd	ɬ	\textlhookt	ʏ	\textscy
ǥ	\textcrg	ɭ	\textlhtlongi	ˌ	\textsecstress
ħ	\textcrh	ʯ	\textlhtlongy	ь	\textsoftsign
ƚ	\textcrinvglotstop	ɾ	\textlonglegr	ʗ	\textstretchc
ƛ	\textcrlambda	˂	\textlptr	ʦ	\texttctclig
ƻ	\textcrtwo	ɱ	\textltailm	ʧ	\textteshlig
ȼ	\textctc	ɲ	\textltailn	θ	\texttheta
ȡ	\textctd	ɬ	\textltilde	þ	\textthorn
ȡʑ	\textctdctzlig	ɮ	\textlyoghlig	ǀ	\texttoneletterstem
ʃ	\textctesh	ʝ	\textObardotlessj	ʦ	\texttslig
ʝ	\textctj	ɮ	\textOlyoghlig	ɐ	\textturna
ɳ	\textctn	ω	\textomega	œ	\textturncelig
ʇ	\textctt	ˠ	\textopencorner	ɥ	\textturnh
ʇɕ	\textcttctclig	ɔ	\textopeno	ʞ	\textturnk
ʒ	\textctyogh	˩	\textpalhook	ʟ	\textturnlonglegr
ʑ	\textctz	ɸ	\textphi	ɯ	\textturnm
ʤ	\textdctzlig	ǀ	\textpipe	ɯɻ	\textturnmrleg
ǯ	\textdoublebaresh	ˈ	\textprimstress	ɹ	\textturnr
ǂ	\textdoublebarpipe	ˀ	\textraiseglotstop	ɻ	\textturnrrtail
≠	\textdoublebarslash	ɺ	\textraisevibyi	ɒ	\textturnscripta
‖	\textdoublepipe	ɤ	\textramshorns	ɬ	\textturnt
‖	\textdoublevertline	ʻ	\textrevapostrophe	ʌ	\textturnv
↓	\textdownstep	ɘ	\textreve	ʍ	\textturnw
ǧ	\textdyoghlig	ɜ	\textrevepsilon	ʎ	\textturny
ʤ	\textdzlig	ʕ	\textrevglotstop	ʊ	\textupsilon
ɛ	\textepsilon	ʢ	\textrevyogh	↑	\textupstep
ʃ	\textesh	ɜʴ	\textrhookrevepsilon	ǀ	\textvertline
ɾ	\textfishhookr	ɚ	\textrhookschwa	ɺ	\textvibyi
g	\textg	˞	\textrhoticity	ɻ	\textvibyy
ɣ	\textgamma	˃	\textrptr	ƿ	\textwynn
↘	\textglobfall	ɖ	\textrtaild	ʒ	\textyogh

(continued on next page)

(continued from previous page)

↗	\textglobrise	ɭ	\textrtaill

tipa defines shortcut characters for many of the above. It also defines a command \tone for denoting tone letters (pitches). See the tipa documentation for more information.

Table 10: wsuipa Phonetic Symbols

ɤ	\babygamma	ŋ	\eng	ɱ	\labdentalnas	ə	\schwa
ƀ	\barb	ɚ	\er	ɬ	\latfric	ɪ	\sci
đ	\bard	ʃ	\esh	ɯ̨	\legm	ɴ	\scn
ɨ	\bari	ð	\eth	ɾ̠	\legr	ʀ	\scr
ɬ	\barl	ɾ	\flapr	ɮ	\lz	ɑ	\scripta
ɵ	\baro	ʔ	\glotstop	ɑ	\nialpha	ɡ	\scriptg
ᵽ	\barp	ɓ	\hookb	β	\nibeta	ʊ	\scriptv
ɨ	\barsci	ɗ	\hookd	χ	\nichi	ᴜ	\scu
ʉ	\barscu	ɠ	\hookg	ɛ	\niepsilon	ʏ	\scy
ʉ	\baru	ɦ	\hookh	ɣ	\nigamma	ƀ	\slashb
⊙	\clickb	ɧ	\hookheng	ɩ	\niiota	ȼ	\slashc
ǀ	\clickc	ɜ	\hookrevepsilon	ƛ	\nilambda	đ	\slashd
ǂ	\clickt	ʰʊ	\hv	ω	\niomega	ɨ	\slashu
ɷ	\closedniomega	ɐ	\inva	ɸ	\niphi	ɗ	\taild
ɞ	\closedrevepsilon	ʃ	\invf	σ	\nisigma	ɻ	\tailinvr
ƀ	\crossb	ʖ	\invglotstop	θ	\nitheta	ɭ	\taill
đ	\crossd	ɥ	\invh	ʊ	\niupsilon	ɳ	\tailn
ħ	\crossh	ɹ	\invlegr	ɲ	\nj	ɽ	\tailr
ƛ	\crossnilambda	ɯ	\invm	œ	\oo	ʂ	\tails
ȼ	\curlyc	ɹ	\invr	ɔ	\openo	ʈ	\tailt
ʃ	\curlyesh	ʁ	\invscr	ɘ	\reve	ʐ	\tailz
ʒ	\curlyyogh	ɒ	\invscripta	ʕ	\reveject	ʧ	\tesh
ʐ	\curlyz	ʌ	\invv	ɜ	\revepsilon	þ	\thorn
ɨ	\dlbari	ʍ	\invw	ʕ	\revglotstop	ƭ	\tildel
ʤ	\dz	ʎ	\invy	ᴅ	\scd	ʒ	\yogh
ʔ	\ejective	ɣ	\ipagamma	ɢ	\scg		

Table 11: phonetic Phonetic Symbols

ɟ	\barj	ɾ	\flap	ɨ	\ibar	ɒ	\rotvara	ɪ	\vari
ƛ	\barlambda	ʔ	\glottal	ɔ	\openo	ʍ	\rotw	ɷ	\varomega
ɱ	\emgma	ʙ	\hausaB	ħ	\planck	ʎ	\roty	ɔ	\varopeno
ŋ	\engma	ɓ	\hausab	ʌ	\pwedge	ə	\schwa	ʏ	\vod
ɲ	\enya	ɗ	\hausad	ᗡ	\revD	þ	\thorn	ɦ	\voicedh
ɛ	\epsi	ᗪ	\hausaD	ɪ	\riota	ʉ	\ubar	ʒ	\yogh
ʃ	\esh	ƙ	\hausak	ɯ	\rotm	ɥ	\udesc		
ð	\eth	Ƙ	\hausaK	ʊ	\rotOmega	ɑ	\vara		
ʄ	\fj	ɗ	\hookd	ɹ	\rotr	ɡ	\varg		

Table 12: Text-mode Accents

Ää	`\"{A}\"{a}`	Àà	`\‘{A}\‘{a}`	Őő	`\H{A}\H{a}`	Ăă	`\u{A}\u{a}`
Áá	`\’{A}\’{a}`	A̲a̲	`\b{A}\b{a}`	Ąą	`\k{A}\k{a}`†	Ǎǎ	`\v{A}\v{a}`
Ȧȧ	`\.{A}\.{a}`	Ąą	`\c{A}\c{a}`	Åå	`\r{A}\r{a}`	Ãã	`\~{A}\~{a}`
Āā	`\={A}\={a}`	Ạạ	`\d{A}\d{a}`	A͡a͡	`\t{A}\t{a}`		
Ââ	`\^{A}\^{a}`	Ää	`\G{A}\G{a}`‡	Ää	`\U{A}\U{a}`‡		

Ââ `\newtie{A}\newtie{a}`* Ⓐⓐ `\textcircled{A}\textcircled{a}`

* Requires the textcomp package.

† Not available in the OT1 font encoding. Use the fontenc package to select an alternate font encoding, such as T1.

‡ Requires the T4 font encoding, provided by the fc package.

Also note the existence of `\i` and `\j`, which produce dotless versions of “i” and “j” (viz., “ı” and “ȷ”). These are useful when the accent is supposed to replace the dot. For example, “`na\"{\i}ve`” produces a correct “naïve”, while “`na\"{i}ve`” would yield the rather odd-looking “naïve”. (“`na\"{i}ve`” *does* work in encodings other than OT1, however.)

Table 13: tipa Text-mode Accents

Á́á́	`\textacutemacron{A}\textacutemacron{a}`
Á̫á̫	`\textacutewedge{A}\textacutewedge{a}`
A̪a̪	`\textadvancing{A}\textadvancing{a}`
A͡a͡	`\textbottomtiebar{A}\textbottomtiebar{a}`
Ă̽ă̽	`\textbrevemacron{A}\textbrevemacron{a}`
Ãã	`\textcircumacute{A}\textcircumacute{a}`
Å̊å̊	`\textcircumdot{A}\textcircumdot{a}`
Ȧ́ȧ́	`\textdotacute{A}\textdotacute{a}`
Ḁ̆ḁ̆	`\textdotbreve{A}\textdotbreve{a}`
Ḁ̆ḁ̆	`\textdotbreve{A}\textdotbreve{a}`
Ȁȁ	`\textdoublegrave{A}\textdoublegrave{a}`
Ä̋ä̋	`\textdoublevbaraccent{A}\textdoublevbaraccent{a}`
Ȃȃ	`\textgravecircum{A}\textgravecircum{a}`
Ȧ̀ȧ̀	`\textgravedot{A}\textgravedot{a}`
Ā̀ā̀	`\textgravemacron{A}\textgravemacron{a}`
Ằằ	`\textgravemid{A}\textgravemid{a}`
A̺a̺	`\textinvsubbridge{A}\textinvsubbridge{a}`
Ạạ	`\textlowering{A}\textlowering{a}`
Á̵á̵	`\textmidacute{A}\textmidacute{a}`
Ǎ̵ǎ̵	`\textovercross{A}\textovercross{a}`

(continued on next page)

(continued from previous page)

A͝a͝	`\textoverw{A}\textoverw{a}`
A̡a̡	`\textpolhook{A}\textpolhook{a}`
A͡a͡	`\textraising{A}\textraising{a}`
A͢a͢	`\textretracting{A}\textretracting{a}`
A͞a͞	`\textringmacron{A}\textringmacron{a}`
A͡a	`\textroundcap{A}\textroundcap{a}`
A͡a	`\textseagull{A}\textseagull{a}`
A̗a̗	`\textsubacute{A}\textsubacute{a}`
A̯a̯	`\textsubarch{A}\textsubarch{a}`
A̲a̲	`\textsubbar{A}\textsubbar{a}`
A̪a̪	`\textsubbridge{A}\textsubbridge{a}`
A̭a̭	`\textsubcircum{A}\textsubcircum{a}`
Ạạ	`\textsubdot{A}\textsubdot{a}`
A̖a̖	`\textsubgrave{A}\textsubgrave{a}`
A̩a̩	`\textsublhalfring{A}\textsublhalfring{a}`
A̟a̟	`\textsubplus{A}\textsubplus{a}`
A̹a̹	`\textsubrhalfring{A}\textsubrhalfring{a}`
Ḁḁ	`\textsubring{A}\textsubring{a}`
A̺a̺	`\textsubsquare{A}\textsubsquare{a}`
A̰a̰	`\textsubtilde{A}\textsubtilde{a}`
A̤a̤	`\textsubumlaut{A}\textsubumlaut{a}`
A̬a̬	`\textsubw{A}\textsubw{a}`
A̬a̬	`\textsubwedge{A}\textsubwedge{a}`
A̸a̸	`\textsuperimposetilde{A}\textsuperimposetilde{a}`
A᷎a᷎	`\textsyllabic{A}\textsyllabic{a}`
Ã̇ã̇	`\texttildedot{A}\texttildedot{a}`
A͡a	`\texttoptiebar{A}\texttoptiebar{a}`
A̍a̍	`\textvbaraccent{A}\textvbaraccent{a}`

tipa defines shortcut sequences for many of the above. See the tipa documentation for more information.

Table 14: wsuipa Text-mode Accents

A̪a̪	`\dental{A}\dental{a}`
A̺a̺	`\underarch{A}\underarch{a}`

Table 15: phonetic Text-mode Accents

A͡a	\hill{A}\hill{a}	A꞉a	\rc{A}\rc{a}	A͜a	\ut{A}\ut{a}
A͘a	\od{A}\od{a}	A̤a	\syl{A}\syl{a}		
Å̂a	\ohill{A}\ohill{a}	A̤a	\td{A}\td{a}		

The phonetic package provides a few additional macros for linguistic accents. \acbar and \acarc compose characters with multiple accents; for example, \acbar{\'}{a} produces "á" and \acarc{\"}{e} produces "ë". \labvel joins two characters with an arc: \labvel{mn} → "m͡n". \upbar is intended to go between characters as in "x\upbar{}y'' → "x|y". Lastly, \uplett behaves like \textsuperscript but uses a smaller font. Contrast "p\uplett{h}'' → "pʰ" with "p^{h}'' → "pʰ".

Table 16: wsuipa Diacritics

ˁ	\ain	˂	\leftp	˚	\overring	ˈ	\stress	ˬ	\underwedge
˥	\corner	ˣ	\leftt	˛	\polishhook	ˌ	\syllabic	˄	\upp
˅	\downp	ː	\length	˃	\rightp	¨	\underdots	˩	\upt
˧	\downt	~	\midtilde	ˢ	\rightt	̥	\underring		
ˑ	\halflength	˓	\open	ˌ	\secstress	̰	\undertilde		

The wsuipa package defines all of the above as ordinary characters, not as accents. However, it does provide \diatop and \diaunder commands, which are used to compose diacritics with other characters. For example, \diatop[\overring|a] produces "å", and \diaunder[\underdots|a] produces "a̤". See the wsuipa documentation for more information.

Table 17: textcomp Diacritics

˝	\textacutedbl	ˇ	\textasciicaron	¯	\textasciimacron
´	\textasciiacute	¨	\textasciidieresis	‵	\textgravedbl
˘	\textasciibreve	`	\textasciigrave		

The textcomp package defines all of the above as ordinary characters, not as accents.

Table 18: textcomp Currency Symbols

฿	\textbaht	$	\textdollar	₲	\textguarani	₩	\textwon
¢	\textcent	$	\textdollaroldstyle	£	\textlira	¥	\textyen
¢	\textcentoldstyle	₫	\textdong	₦	\textnaira		
₡	\textcolonmonetary	€	\texteuro	₱	\textpeso		
¤	\textcurrency	f	\textflorin	£	\textsterling		

Table 19: marvosym Currency Symbols

ℛ	\Denarius	€	\EUR	€	\EURdig	€	\EURtm	℔	\Pfund
@	\Ecommerce	€	\EURcr	€	\EURhv	$	\EyesDollar	℔	\Shilling

The different euro signs are meant to be compatible with different fonts—Courier (\EURcr), Helvetica (\EURhv), Times (\EURtm), and the marvosym digits listed in Table 134 (\EURdig).

Table 20: wasysym Currency Symbols

¢	\cent	¤	\currency

Table 21: eurosym Euro Signs

€	\geneuro	€	\geneuronarrow	€	\geneurowide	€	\officialeuro

\euro is automatically mapped to one of the above—by default, \officialeuro—based on a eurosym package option. See the eurosym documentation for more information. The \geneuro... characters are generated from the current body font's "C" character and therefore may not appear exactly as shown.

Table 22: textcomp Legal Symbols

Ⓟ	\textcircledP	©	Ⓒ	\textcopyright	℠	℠	\textservicemark
ↄ	\textcopyleft	®	Ⓡ	\textregistered	™	™	\texttrademark

Where two symbols are present, the left one is the "faked" symbol that LaTeX 2_ε provides by default, and the right one is the "true" symbol that textcomp makes available.

See http://www.tex.ac.uk/cgi-bin/texfaq2html?label=tradesyms for solutions to common problems that occur when using these symbols (e.g., getting a "ⓡ" when you expected to get a "Ⓡ").

Table 23: textcomp Old-style Numerals

0	\textzerooldstyle	4	\textfouroldstyle	8	\texteightoldstyle
1	\textoneoldstyle	5	\textfiveoldstyle	9	\textnineoldstyle
2	\texttwooldstyle	6	\textsixoldstyle		
3	\textthreeoldstyle	7	\textsevenoldstyle		

Rather than use the bulky \textoneoldstyle, \texttwooldstyle, etc. commands shown above, consider using \oldstylenums{...} to typeset an old-style number.

Table 24: Miscellaneous textcomp Symbols

⁎	\textasteriskcentered		ª	a	\textordfeminine
‖	\textbardbl		º	o	\textordmasculine
◯	\textbigcircle		¶		\textparagraph
ƀ	\textblank		·		\textperiodcentered
¦	\textbrokenbar		‱		\textpertenthousand
•	\textbullet		‰		\textperthousand
†	\textdagger		¶		\textpilcrow
‡	\textdaggerdbl		'		\textquotesingle
⸗	\textdblhyphen		‚		\textquotestraightbase
=	\textdblhyphenchar		„		\textquotestraightdblbase
⁒	\textdiscount		℞		\textrecipe
℮	\textestimated		※		\textreferencemark
⸘	\textinterrobang		§		\textsection
¡	\textinterrobangdown		—		\textthreequartersemdash
♪	\textmusicalnote		~		\texttildelow
№	\textnumero		–		\texttwelveudash
○	\textopenbullet				

Where two symbols are present, the left one is the "faked" symbol that LaTeX 2_ε provides by default, and the right one is the "true" symbol that textcomp makes available.

Table 25: Miscellaneous wasysym Text-mode Symbols

‰ \permil

Table 26: $\mathcal{AMS}$ Commands Defined to Work in Both Math and Text Mode

✓ \checkmark ® \circledR ✠ \maltese

3 Mathematical symbols

Most, but not all, of the symbols in this section are math-mode only. That is, they yield a "`Missing $ inserted`" error message if not used within `$...$`, `\[...\]`, or another math-mode environment. Operators marked as "variable-sized" are taller in displayed formulas, shorter in in-text formulas, and possibly shorter still when used in various levels of superscripts or subscripts.

Alphanumeric symbols (e.g., "$\mathscr{L}$" and "$\mathbb{Z}$") are usually produced using one of the math alphabets in Table 135 rather than with an explicit symbol command. Look there first if you need a symbol for a transform, number set, or some other alphanumeric.

Although there have been many requests on `comp.text.tex` for a contradiction symbol, the ensuing discussion invariably reveals innumerable ways to represent contradiction in a proof, including "↯" (`\blitza`), "⇒⇐" (`\Rightarrow\Leftarrow`), "⊥" (`\bot`), "↮" (`\nleftrightarrow`), and "※" (`\textreferencemark`). Because of the lack of notational consensus, it is probably better to spell out "Contradiction!" than to use a symbol for this purpose. Similarly, discussions on `comp.text.tex` have revealed that there are a variety of ways to indicate the mathematical notion of "is defined as". Common candidates include "≜" (`\triangleq`), "≡" (`\equiv`), "≔" (`\coloneqq`), and "$\stackrel{\text{def}}{=}$" (`\stackrel{\text{def}}{=}`). See also the example of `\equalsfill` on page 61.

Table 27: Binary Operators

∐	`\amalg`	∪	`\cup`	⊕	`\oplus`	×	`\times`
∗	`\ast`	†	`\dagger`	⊘	`\oslash`	◁	`\triangleleft`
◯	`\bigcirc`	‡	`\ddagger`	⊗	`\otimes`	▷	`\triangleright`
▽	`\bigtriangledown`	◇	`\diamond`	±	`\pm`	⊴	`\unlhd`*
△	`\bigtriangleup`	÷	`\div`	▷	`\rhd`*	⊵	`\unrhd`*
•	`\bullet`	◁	`\lhd`*	\	`\setminus`	⊎	`\uplus`
∩	`\cap`	∓	`\mp`	⊓	`\sqcap`	∨	`\vee`
·	`\cdot`	⊙	`\odot`	⊔	`\sqcup`	∧	`\wedge`
∘	`\circ`	⊖	`\ominus`	⋆	`\star`	≀	`\wr`

* Not predefined in LaTeX 2ε. Use one of the packages latexsym, amsfonts, amssymb, txfonts, pxfonts, or wasysym.

Table 28: $\mathcal{AMS}$ Binary Operators

⊼	`\barwedge`	⊚	`\circledcirc`	⊺	`\intercal`
⊡	`\boxdot`	⊝	`\circleddash`	⋋	`\leftthreetimes`
⊟	`\boxminus`	⋓	`\Cup`	⋉	`\ltimes`
⊞	`\boxplus`	⋎	`\curlyvee`	⋌	`\rightthreetimes`
⊠	`\boxtimes`	⋏	`\curlywedge`	⋊	`\rtimes`
⋒	`\Cap`	⋇	`\divideontimes`	∖	`\smallsetminus`
⋅	`\centerdot`	∔	`\dotplus`	⊻	`\veebar`
⊛	`\circledast`	⩚	`\doublebarwedge`		

Table 29: stmaryrd Binary Operators

⟠	\baro	⫼	\interleave	⊛	\varoast
⦸	\bbslash	⪡	\leftslice	⊍	\varobar
⅋	\binampersand	⩛	\merge	⦸	\varobslash
⋇	\bindnasrepma	⨪	\minuso	⊚	\varocircle
⧆	\boxast	⫫	\moo	⊙	\varodot
⧈	\boxbar	⨣	\nplus	⊖	\varogreaterthan
⧉	\boxbox	⦶	\obar	⊖	\varolessthan
⧅	\boxbslash	⧠	\oblong	⊖	\varominus
⧇	\boxcircle	⦸	\obslash	⊕	\varoplus
⊡	\boxdot	⧁	\ogreaterthan	⊘	\varoslash
☐	\boxempty	⧀	\olessthan	⊗	\varotimes
⧄	\boxslash	⩒	\ovee	⩒	\varovee
⤈	\curlyveedownarrow	⬯	\owedge	⬯	\varowedge
⤉	\curlyveeuparrow	⧐	\rightslice	⨯	\vartimes
⤋	\curlywedgedownarrow	⫽	\sslash	⟙	\Ydown
⤊	\curlywedgeuparrow	⫴	\talloblong	⪡	\Yleft
⧵	\fatbslash	◯	\varbigcirc	⪢	\Yright
⨟	\fatsemi	⋎	\varcurlyvee	⟘	\Yup
⫻	\fatslash	⋏	\varcurlywedge		

Table 30: wasysym Binary Operators

◁	\lhd	○	\ocircle	▶	\RHD	⊵	\unrhd
◀	\LHD	▷	\rhd	⊴	\unlhd		

Table 31: txfonts/pxfonts Binary Operators

⦶	\circledbar	⬯	\circledwedge	○	\medcirc
⦸	\circledbslash	⅋	\invamp	⊞	\sqcapplus
⩒	\circledvee	●	\medbullet	⊎	\sqcupplus

Table 32: mathabx Binary Operators

∗	\ast	⋏	\curlywedge	⊓	\sqcap
✻	\Asterisk	⨪	\divdot	⊔	\sqcup
⩛	\barwedge	⨳	\divideontimes	⩏	\sqdoublecap
★	\bigstar	∸	\dotdiv	⩐	\sqdoublecup
✶	\bigvarstar	∔	\dotplus	▫	\square
◆	\blackdiamond	⨯	\dottimes	⊞	\squplus
∩	\cap	⩛	\doublebarwedge	·	\udot
⨣	\circplus	⋒	\doublecap	⊎	\uplus
⊛	\coasterisk	⋓	\doublecup	⋆	\varstar
✲	\coAsterisk	⋉	\ltimes	∨	\vee
✳	\convolution	⨤	\pluscirc	⩡	\veebar
∪	\cup	⋊	\rtimes	⩢	\veedoublebar
⋎	\curlyvee	▪	\sqbullet	∧	\wedge

Many of the above glyphs go by multiple names. \centerdot is equivalent to \sqbullet, and \ast is equivalent to *. \asterisk produces the same glyph as \ast, but as an ordinary symbol, not a binary operator. Similarly, \bigast produces a large-operator version of the \Asterisk binary operator, and \bigcoast produces a large-operator version of the \coAsterisk binary operator.

Table 33: ulsy Geometric Binary Operators

⦶	\odplus

Table 34: mathabx Geometric Binary Operators

▾	\blacktriangledown	⊞	\boxright	⊖	\ominus
◂	\blacktriangleleft	⧄	\boxslash	⊕	\oplus
▸	\blacktriangleright	⊠	\boxtimes	⊕	\oright
▴	\blacktriangleup	⊟	\boxtop	⊘	\oslash
⊛	\boxasterisk	⧋	\boxtriangleup	⊗	\otimes
⧅	\boxbackslash	□	\boxvoid	⊗	\otop
⊟	\boxbot	⊛	\oasterisk	⧋	\otriangleup
⊡	\boxcirc	⦸	\obackslash	◯	\ovoid
⊛	\boxcoasterisk	⦶	\obot	▿	\smalltriangledown
⊡	\boxdiv	⊙	\ocirc	◃	\smalltriangleleft
⊡	\boxdot	⊛	\ocoasterisk	▹	\smalltriangleright
⊞	\boxleft	⨸	\odiv	▵	\smalltriangleup
⊟	\boxminus	⊙	\odot		
⊞	\boxplus	⊕	\oleft		

Table 35: Variable-sized Math Operators

$\bigcap$	\bigcap	$\bigotimes$	\bigotimes	$\bigwedge$	\bigwedge	$\prod$	\prod
$\bigcup$	\bigcup	$\bigsqcup$	\bigsqcup	$\coprod$	\coprod	$\sum$	\sum
$\bigodot$	\bigodot	$\biguplus$	\biguplus	$\int$	\int		
$\bigoplus$	\bigoplus	$\bigvee$	\bigvee	$\oint$	\oint		

TABLE 36: $\mathcal{AMS}$ Variable-sized Math Operators

$\int\cdots\int$ $\int\cdots\int$	\idotsint	$\iiint$ $\iiint$	\iiint
$\iiiint$ $\iiiint$	\iiiint	$\iint$ $\iint$	\iint

TABLE 37: stmaryrd Variable-sized Math Operators

$\bigbox$	\bigbox	$\biginterleave$	\biginterleave	$\bigsqcap$	\bigsqcap
$\bigcurlyvee$	\bigcurlyvee	$\bignplus$	\bignplus	$\bigtriangledown$	\bigtriangledown
$\bigcurlywedge$	\bigcurlywedge	$\bigparallel$	\bigparallel	$\bigtriangleup$	\bigtriangleup

TABLE 38: wasysym Variable-sized Math Operators

$\iiint$ $\iiint$	\iiint	$\oiint$ $\oiint$	\oiint	$\varointclockwise$ $\varointclockwise$	\varoint
$\iint$ $\iint$	\iint	$\varint$ $\varint$	\varint		

TABLE 39: mathabx Variable-sized Math Operators

$\bigcurlyvee$ $\bigcurlyvee$	\bigcurlyvee	$\bigboxslash$ $\bigboxslash$	\bigboxslash	$\bigoright$ $\bigoright$	\bigoright
$\bigsqcap$ $\bigsqcap$	\bigsqcap	$\bigboxtimes$ $\bigboxtimes$	\bigboxtimes	$\bigoslash$ $\bigoslash$	\bigoslash
$\bigcurlywedge$ $\bigcurlywedge$	\bigcurlywedge	$\bigboxtop$ $\bigboxtop$	\bigboxtop	$\bigotop$ $\bigotop$	\bigotop
$\bigboxasterisk$ $\bigboxasterisk$	\bigboxasterisk	$\bigboxtriangleup$ $\bigboxtriangleup$	\bigboxtriangleup	$\bigotriangleup$ $\bigotriangleup$	\bigotriangleup
$\bigboxbackslash$ $\bigboxbackslash$	\bigboxbackslash	$\bigboxvoid$ $\bigboxvoid$	\bigboxvoid	$\bigovoid$ $\bigovoid$	\bigovoid
$\bigboxbot$ $\bigboxbot$	\bigboxbot	$\bigcomplementop$ $\bigcomplementop$	\bigcomplementop	$\bigplus$ $\bigplus$	\bigplus
$\bigboxcirc$ $\bigboxcirc$	\bigboxcirc	$\bigoasterisk$ $\bigoasterisk$	\bigoasterisk	$\bigsquplus$ $\bigsquplus$	\bigsquplus
$\bigboxcoasterisk$ $\bigboxcoasterisk$	\bigboxcoasterisk	$\bigobackslash$ $\bigobackslash$	\bigobackslash	$\bigtimes$ $\bigtimes$	\bigtimes
$\bigboxdiv$ $\bigboxdiv$	\bigboxdiv	$\bigobot$ $\bigobot$	\bigobot	$\iiint$ $\iiint$	\iiint
$\bigboxdot$ $\bigboxdot$	\bigboxdot	$\bigocirc$ $\bigocirc$	\bigocirc	$\iint$ $\iint$	\iint
$\bigboxleft$ $\bigboxleft$	\bigboxleft	$\bigocoasterisk$ $\bigocoasterisk$	\bigocoasterisk	$\int$ $\int$	\int
$\bigboxminus$ $\bigboxminus$	\bigboxminus	$\bigodiv$ $\bigodiv$	\bigodiv	$\oiint$ $\oiint$	\oiint
$\bigboxplus$ $\bigboxplus$	\bigboxplus	$\bigoleft$ $\bigoleft$	\bigoleft	$\oint$ $\oint$	\oint
$\bigboxright$ $\bigboxright$	\bigboxright	$\bigominus$ $\bigominus$	\bigominus		

TABLE 40: txfonts/pxfonts Variable-sized Math Operators

⨦	⨁	\bigsqcapplus	∲	∲	\ointclockwise
⨥	⨁	\bigsqcupplus	∳	∳	\ointctrclockwise
⨍	⨍	\fint	⨌	⨌	\sqiiint
∫⋯∫	∫⋯∫	\idotsint	⨏	⨏	\sqiint
⨌	⨌	\iiiint	⨖	⨖	\sqint
∭	∭	\iiint	⨐	⨐	\varoiiintclockwise
∬	∬	\iint	⨑	⨑	\varoiiintctrclockwise
∰	∰	\oiiintclockwise	⨒	⨒	\varoiintclockwise
∰	∰	\oiiintctrclockwise	⨓	⨓	\varoiintctrclockwise
∰	∰	\oiiint	⨍	⨍	\varointclockwise
∯	∯	\oiintclockwise	⨎	⨎	\varointctrclockwise
∯	∯	\oiintctrclockwise	⨉	⨉	\varprod
∯	∯	\oiint			

TABLE 41: esint Variable-sized Math Operators

∫⋯∫	∫⋯∫	\dotsint	∲	∲	\ointclockwise
⨍	⨍	\fint	∳	∳	\ointctrclockwise
⨌	⨌	\iiiint	⨋	⨋	\sqiint
∭	∭	\iiint	⨖	⨖	\sqint
∬	∬	\iint	⨏	⨏	\varoiint
⨗	⨗	\landdownint	∳	∳	\varointclockwise
⨘	⨘	\landupint	∳	∳	\varointctrclockwise
∯	∯	\oiint			

TABLE 42: Binary Relations

$\approx$	\approx	$\equiv$	\equiv	$\perp$	\perp	$\smile$	\smile
$\asymp$	\asymp	$\frown$	\frown	$\prec$	\prec	$\succ$	\succ
$\bowtie$	\bowtie	$\bowtie$	\Join*	$\preceq$	\preceq	$\succeq$	\succeq
$\cong$	\cong	$\mid$	\mid	$\propto$	\propto	$\vdash$	\vdash
$\dashv$	\dashv	$\models$	\models	$\sim$	\sim		
$\doteq$	\doteq	$\parallel$	\parallel	$\simeq$	\simeq		

* Not predefined in LaTeX 2_ε. Use one of the packages latexsym, amsfonts, amssymb, mathabx, txfonts, pxfonts, or wasysym.

TABLE 43: $\mathcal{AMS}$ Binary Relations

$\backepsilon$	\backepsilon	$\multimap$	\multimap	$\therefore$	\therefore
$\backsim$	\backsim	$\pitchfork$	\pitchfork	$\thickapprox$	\thickapprox
$\backsimeq$	\backsimeq	$\precapprox$	\precapprox	$\thicksim$	\thicksim
$\because$	\because	$\preccurlyeq$	\preccurlyeq	$\varpropto$	\varpropto
$\between$	\between	$\precsim$	\precsim	$\Vdash$	\Vdash
$\Bumpeq$	\Bumpeq	$\risingdotseq$	\risingdotseq	$\vDash$	\vDash
$\bumpeq$	\bumpeq	$\shortmid$	\shortmid	$\Vvdash$	\Vvdash
$\circeq$	\circeq	$\shortparallel$	\shortparallel		
$\curlyeqprec$	\curlyeqprec	$\smallfrown$	\smallfrown		
$\curlyeqsucc$	\curlyeqsucc	$\smallsmile$	\smallsmile		
$\doteqdot$	\doteqdot				

TABLE 44: $\mathcal{AMS}$ Negated Binary Relations

$\ncong$	\ncong	$\nshortparallel$	\nshortparallel	$\nVDash$	\nVDash
$\nmid$	\nmid	$\nsim$	\nsim	$\precnapprox$	\precnapprox
$\nparallel$	\nparallel	$\nsucc$	\nsucc	$\precnsim$	\precnsim
$\nprec$	\nprec	$\nsucceq$	\nsucceq	$\succnapprox$	\succnapprox
$\npreceq$	\npreceq	$\nvDash$	\nvDash	$\succnsim$	\succnsim
$\nshortmid$	\nshortmid	$\nvdash$	\nvdash		

TABLE 45: stmaryrd Binary Relations

$\Subset$	\inplus	$\Supset$	\niplus

TABLE 46: wasysym Binary Relations

$\invneg$	\invneg	$\leadsto$	\leadsto	$\propto$	\wasypropto
$\bowtie$	\Join	$\circledast$	\logof		

Table 47: txfonts/pxfonts Binary Relations

⦸	\circledgtr	⋉	\lJoin	×	\opentimes
⦵	\circledless	⋈	\lrtimes	⫫	\Perp
∴≈	\colonapprox	⊸	\multimap	⪷	\preceqq
∷≈	\Colonapprox	⊶	\multimapboth	⪵	\precneqq
∴−	\coloneq	⧟	\multimapbothvert	⋊	\rJoin
∷−	\Coloneq	⧟	\multimapdot	⫟	\strictfi
∷=	\Coloneqq	⧟	\multimapdotboth	⫞	\strictif
∴=	\coloneqq	⧟	\multimapdotbothA	⫠	\strictiff
∷∼	\Colonsim	⧟	\multimapdotbothAvert	⪸	\succeqq
∴∼	\colonsim	⧟	\multimapdotbothB	⪹	\succneqq
−∷	\Eqcolon	⧟	\multimapdotbothBvert	⫽	\varparallel
−∴	\eqcolon	⧟	\multimapdotbothvert	⫻	\varparallelinv
=∴	\eqqcolon	⧟	\multimapdotinv	⫤	\VvDash
=∷	\Eqqcolon	⧟	\multimapinv		
≂	\eqsim	×	\openJoin		

Table 48: txfonts/pxfonts Negated Binary Relations

≇	\napproxeq	⪹̸	\npreccurlyeq	≉	\nthickapprox
≁	\nasymp	⪷̸	\npreceqq	↞̸	\ntwoheadleftarrow
∽̸	\nbacksim	⋨	\nprecsim	↠̸	\ntwoheadrightarrow
≋̸	\nbacksimeq	≄	\nsimeq	∦	\nvarparallel
≎̸	\nbumpeq	⪸̸	\nsuccapprox	⫮	\nvarparallelinv
≏̸	\nBumpeq	⪺̸	\nsucccurlyeq	⊯	\nVdash
≢	\nequiv	⪺̸	\nsucceqq		
⪯̸	\nprecapprox	⋩	\nsuccsim		

Table 49: mathabx Binary Relations

≬	\between	∣	\divides	≓	\risingdotseq
≒	\botdoteq	≒	\doteqdot	≳	\succapprox
≎	\Bumpedeq	≑	\eqbumped	≽	\succcurlyeq
≏	\bumpedeq	≖	\eqcirc	≻	\succdot
≗	\circeq	≔	\eqcolon	≿	\succsim
≔	\coloneq	≒	\fallingdotseq	∴	\therefore
≙	\corresponds	≫	\ggcurly	≐	\topdoteq
≼	\curlyeqprec	≪	\llcurly	⊨	\vDash
≽	\curlyeqsucc	≳	\precapprox	⊩	\Vdash
⫤	\DashV	≼	\preccurlyeq	⊫	\VDash
⫣	\Dashv	≺	\precdot	⊪	\Vvdash
⫦	\dashVv	≾	\precsim		

TABLE 50: mathabx Negated Binary Relations

≉	\napprox	⊥̸	\notperp	⊭	\nvDash
≇	\ncong	⊀	\nprec	⊯	\nVDash
⋜̸	\ncurlyeqprec	⪷̸	\nprecapprox	⊮	\nVdash
⋝̸	\ncurlyeqsucc	⋠	\npreccurlyeq	⊬	\nvdash
⊬	\nDashv	⪯̸	\npreceq	⊮	\nVvash
⊮	\ndashV	⪹	\nprecsim	⪺̸	\precnapprox
⊬	\ndashv	⊁	\nsim	⪵̸	\precneq
⊯	\nDashV	≄	\nsimeq	⪹̸	\precnsim
⊯	\ndashVv	⊁	\nsucc	⪺̸	\succnapprox
≠	\neq	⪸̸	\nsuccapprox	⪶̸	\succneq
≁	\notasymp	⋡	\nsucccurlyeq	⪺̸	\succnsim
∤	\notdivides	⪰̸	\nsucceq		
≢	\notequiv	⪺	\nsuccsim		

The \changenotsign command toggles the behavior of \not to produce either a vertical or a diagonal slash through a binary operator. Thus, "$a \not= b$" can be made to produce either "$a \not= b$" or "$a \neq b$".

TABLE 51: trsym Binary Relations

⊷	\InversTransformHoriz	⊶	\TransformHoriz
⥉	\InversTransformVert	⥈	\TransformVert

TABLE 52: trfsigns Binary Relations

⟃	\dfourier	⟄	\Dfourier
⟞	\fourier	⟝	\Fourier
⧦	\laplace	⧥	\Laplace
⨟	\ztransf	⨠	\Ztransf

TABLE 53: Subset and Superset Relations

⊏	\sqsubset*	⊒	\sqsupseteq	⊃	\supset
⊑	\sqsubseteq	⊂	\subset	⊇	\supseteq
⊐	\sqsupset*	⊆	\subseteq		

* Not predefined in LaTeX 2ε. Use one of the packages latexsym, amsfonts, amssymb, mathabx, txfonts, pxfonts, or wasysym.

TABLE 54: $\mathcal{AMS}$ Subset and Superset Relations

⊈	\nsubseteq	⫅	\subseteqq	⫆	\supsetneqq
⊉	\nsupseteq	⫋	\subsetneq	⫋	\varsubsetneq
⫋	\nsupseteqq	⫋	\subsetneqq	⫋	\varsubsetneqq
⊏	\sqsubset	⋑	\Supset	⫌	\varsupsetneq
⊐	\sqsupset	⫆	\supseteqq	⫌	\varsupsetneqq
⋐	\Subset	⊋	\supsetneq		

TABLE 55: stmaryrd Subset and Superset Relations

⫋	\subsetplus	⫌	\supsetplus
⫅	\subsetpluseq	⫆	\supsetpluseq

TABLE 56: wasysym Subset and Superset Relations

⊏	\sqsubset	⊐	\sqsupset

TABLE 57: txfonts/pxfonts Subset and Superset Relations

⋢	\nsqsubset	⋣	\nsqsupseteq	⫐	\nSupset
⫋	\nsqsubseteq	⫉	\nSubset		
⋣	\nsqsupset	⫋	\nsubseteqq		

TABLE 58: mathabx Subset and Superset Relations

⋤	\nsqsubset	⊅	\nsupset	⊒	\sqsupseteq	⊇	\supseteq
⫉	\nsqSubset	⫐	\nSupset	⫒	\sqsupseteqq	⫆	\supseteqq
⋥	\nsqsubseteq	⊉	\nsupseteq	⋣	\sqsupsetneq	⊋	\supsetneq
⫋	\nsqsubseteqq	⫉	\nsupseteqq	⫌	\sqsupsetneqq	⫌	\supsetneqq
⋣	\nsqsupset	⊏	\sqsubset	⊂	\subset	⫋	\varsqsubsetneq
⫉	\nsqSupset	⫍	\sqSubset	⋐	\Subset	⫋	\varsqsubsetneqq
⋥	\nsqsupseteq	⊑	\sqsubseteq	⊆	\subseteq	⫌	\varsqsupsetneq
⫋	\nsqsupseteqq	⫏	\sqsubseteqq	⫅	\subseteqq	⫌	\varsqsupsetneqq
⊄	\nsubset	⋤	\sqsubsetneq	⊊	\subsetneq	⊊	\varsubsetneq
⫋	\nSubset	⫋	\sqsubsetneqq	⫋	\subsetneqq	⫋	\varsubsetneqq
⋢	\nsubseteq	⫉	\sqSupset	⊃	\supset	⊋	\varsupsetneq
⫋	\nsubseteqq	⊐	\sqsupset	⫐	\Supset	⫌	\varsupsetneqq

TABLE 59: Inequalities

≥	\geq	≫	\gg	≤	\leq	≪	\ll	≠	\neq

TABLE 60: $\mathcal{AMS}$ Inequalities

⩾	\eqslantgtr	⋛	\gtrless	⪇	\lneq
⩽	\eqslantless	≷	\gtrsim	≨	\lneqq
≧	\geqq	≩	\gvertneqq	⋦	\lnsim
⩾	\geqslant	≦	\leqq	≨	\lvertneqq
⋙	\ggg	⩽	\leqslant	≱	\ngeq
⪊	\gnapprox	≲	\lessapprox	≱	\ngeqq
≩	\gneq	≐	\lessdot	⩾̸	\ngeqslant
≩	\gneqq	⋚	\lesseqgtr	≯	\ngtr
⋧	\gnsim	⪋	\lesseqqgtr	≰	\nleq
≳	\gtrapprox	≶	\lessgtr	≰	\nleqq
⋗	\gtrdot	≲	\lesssim	⩽̸	\nleqslant
⋛	\gtreqless	⋘	\lll	≮	\nless
⪌	\gtreqqless	⪉	\lnapprox		

Table 61: wasysym Inequalities

$\gtrapprox$	\apprge	$\lessapprox$	\apprle

Table 62: txfonts/pxfonts Inequalities

$\ggg$	\ngg	$\gtrsim$	\ngtrsim	$\lesssim$	\nlesssim
$\gtrapprox$	\ngtrapprox	$\lessapprox$	\nlessapprox	$\lll$	\nll
$\gtrless$	\ngtrless	$\lessgtr$	\nlessgtr		

Table 63: mathabx Inequalities

$\geqslant$	\eqslantgtr	$\gtreqless$	\gtreqless	$\lesssim$	\lesssim	$\ngtr$	\ngtr
$\leqslant$	\eqslantless	$\gtreqqless$	\gtreqqless	$\ll$	\ll	$\ngtrapprox$	\ngtrapprox
$\geqslant$	\geq	$\gtrless$	\gtrless	$\lll$	\lll	$\ngtrsim$	\ngtrsim
$\geqq$	\geqq	$\gtrsim$	\gtrsim	$\lnapprox$	\lnapprox	$\nleq$	\nleq
$\gg$	\gg	$\gvertneqq$	\gvertneqq	$\lneq$	\lneq	$\nleqq$	\nleqq
$\ggg$	\ggg	$\leqslant$	\leq	$\lneqq$	\lneqq	$\nless$	\nless
$\gnapprox$	\gnapprox	$\leqq$	\leqq	$\lnsim$	\lnsim	$\nlessapprox$	\nlessapprox
$\gneq$	\gneq	$\lessapprox$	\lessapprox	$\lvertneqq$	\lvertneqq	$\nlesssim$	\nlesssim
$\gneqq$	\gneqq	$\lessdot$	\lessdot	$\neqslantgtr$	\neqslantgtr	$\nvargeq$	\nvargeq
$\gnsim$	\gnsim	$\lesseqgtr$	\lesseqgtr	$\neqslantless$	\neqslantless	$\nvarleq$	\nvarleq
$\gtrapprox$	\gtrapprox	$\lesseqqgtr$	\lesseqqgtr	$\ngeq$	\ngeq	$\vargeq$	\vargeq
$\gtrdot$	\gtrdot	$\lessgtr$	\lessgtr	$\ngeqq$	\ngeqq	$\varleq$	\varleq

mathabx defines \leqslant and \le as synonyms for \leq, \geqslant and \ge as synonyms for \geq, \nleqslant as a synonym for \nleq, and \ngeqslant as a synonym for \ngeq.

Table 64: $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ Triangle Relations

◀	\blacktriangleleft	$\ntrianglelefteq$	\ntrianglelefteq	$\trianglelefteq$	\trianglelefteq	$\vartriangleleft$	\vartriangleleft
▶	\blacktriangleright	$\ntriangleright$	\ntriangleright	$\triangleq$	\triangleq	$\vartriangleright$	\vartriangleright
$\ntriangleleft$	\ntriangleleft	$\ntrianglerighteq$	\ntrianglerighteq	$\trianglerighteq$	\trianglerighteq		

Table 65: stmaryrd Triangle Relations

$\trianglelefteqslant$	\trianglelefteqslant	$\trianglerighteqslant$	\trianglerighteqslant
$\ntrianglelefteqslant$	\ntrianglelefteqslant	$\ntrianglerighteqslant$	\ntrianglerighteqslant

Table 66: mathabx Triangle Relations

$\ntriangleleft$	\ntriangleleft	$\ntrianglerighteq$	\ntrianglerighteq	$\triangleright$	\triangleright	$\vartriangleright$	\vartriangleright
$\ntrianglelefteq$	\ntrianglelefteq	$\triangleleft$	\triangleleft	$\trianglerighteq$	\trianglerighteq		
$\ntriangleright$	\ntriangleright	$\trianglelefteq$	\trianglelefteq	$\vartriangleleft$	\vartriangleleft		

Table 67: Arrows

⇓	\Downarrow	⟵	\longleftarrow	↖	\nwarrow
↓	\downarrow	⟸	\Longleftarrow	⇒	\Rightarrow
↩	\hookleftarrow	⟷	\longleftrightarrow	→	\rightarrow
↪	\hookrightarrow	⟺	\Longleftrightarrow	↘	\searrow
⇝	\leadsto*	⟼	\longmapsto	↙	\swarrow
←	\leftarrow	⟹	\Longrightarrow	↑	\uparrow
⇐	\Leftarrow	⟶	\longrightarrow	⇑	\Uparrow
⇔	\Leftrightarrow	↦	\mapsto	↕	\updownarrow
↔	\leftrightarrow	↗	\nearrow†	⇕	\Updownarrow

* Not predefined in LaTeX 2ε. Use one of the packages latexsym, amsfonts, amssymb, txfonts, pxfonts, or wasysym.

† See the note beneath Table 112 for information about how to put a diagonal arrow across a mathematical expression (as in "$\not\nabla \cdot \vec{B}^{0}$") .

Table 68: Harpoons

↽	\leftharpoondown	⇁	\rightharpoondown	⇌	\rightleftharpoons
↼	\leftharpoonup	⇀	\rightharpoonup		

Table 69: textcomp Text-mode Arrows

↓	\textdownarrow	→	\textrightarrow
←	\textleftarrow	↑	\textuparrow

Table 70: $\mathcal{AMS}$ Arrows

↺	\circlearrowleft	⇇	\leftleftarrows	⇄	\rightleftarrows
↻	\circlearrowright	⇆	\leftrightarrows	⇉	\rightrightarrows
↶	\curvearrowleft	↭	\leftrightsquigarrow	⇝	\rightsquigarrow
↷	\curvearrowright	⇚	\Lleftarrow	↱	\Rsh
⇠	\dashleftarrow	↫	\looparrowleft	↞	\twoheadleftarrow
⇢	\dashrightarrow	↬	\looparrowright	↠	\twoheadrightarrow
⇊	\downdownarrows	↰	\Lsh	⇈	\upuparrows
↢	\leftarrowtail	↣	\rightarrowtail		

Table 71: $\mathcal{AMS}$ Negated Arrows

⇍	\nLeftarrow	⇎	\nLeftrightarrow	⇏	\nRightarrow
↚	\nleftarrow	↮	\nleftrightarrow	↛	\nrightarrow

Table 72: $\mathcal{AMS}$ Harpoons

⇃	\downharpoonleft	⇋	\leftrightharpoons	↿	\upharpoonleft
⇂	\downharpoonright	⇌	\rightleftharpoons	↾	\upharpoonright

Table 73: stmaryrd Arrows

⇜	\leftarrowtriangle	⇼	\Mapsfrom	←	\shortleftarrow
⇿	\leftrightarroweq	↤	\mapsfrom	→	\shortrightarrow
⇹	\leftrightarrowtriangle	⇻	\Mapsto	↑	\shortuparrow
↯	\lightning	↗	\nnearrow	↘	\ssearrow
⟸	\Longmapsfrom	↖	\nnwarrow	↙	\sswarrow
⟻	\longmapsfrom	⇾	\rightarrowtriangle		
⟾	\Longmapsto	↓	\shortdownarrow		

Table 74: txfonts/pxfonts Arrows

⇐⊡	\boxdotLeft	⊙→	\circleddotright	←◇	\Diamondleft
←⊡	\boxdotleft	←○	\circleleft	◇→	\Diamondright
⊡→	\boxdotright	○→	\circleright	◇⇛	\DiamondRight
⊡⇛	\boxdotRight	⇠	\dashleftrightarrow	⇜	\leftsquigarrow
⇐□	\boxLeft	⇐◈	\DiamonddotLeft	⇗	\Nearrow
←□	\boxleft	←◈	\Diamonddotleft	⇖	\Nwarrow
□→	\boxright	◈→	\Diamonddotright	⇛	\Rrightarrow
□⇛	\boxRight	◈⇛	\DiamonddotRight	⇘	\Searrow
←⊙	\circleddotleft	⇐◇	\DiamondLeft	⇙	\Swarrow

Table 75: mathabx Arrows

↺	\circlearrowleft	←	\leftarrow	↖	\nwarrow
↻	\circlearrowright	⇇	\leftleftarrows	↾	\restriction
↶	\curvearrowbotleft	↔	\leftrightarrow	→	\rightarrow
↭	\curvearrowbotleftright	⇆	\leftrightarrows	⇄	\rightleftarrows
↷	\curvearrowbotright	↭	\leftrightsquigarrow	⇉	\rightrightarrows
↶	\curvearrowleft	↜	\leftsquigarrow	⇝	\rightsquigarrow
↭	\curvearrowleftright	↻	\lefttorightarrow	↩	\righttoleftarrow
↷	\curvearrowright	↫	\looparrowdownleft	↱	\Rsh
↲	\dlsh	↬	\looparrowdownright	↘	\searrow
⇊	\downdownarrows	↫	\looparrowleft	↙	\swarrow
↺	\downtouparrow	↬	\looparrowright	↑↓	\updownarrows
↓↑	\downuparrows	↰	\Lsh	↺	\uptodownarrow
↳	\drsh	↗	\nearrow	↑↑	\upuparrows

Table 76: mathabx Negated Arrows

⇍	\nLeftarrow	↮	\nleftrightarrow	↛	\nrightarrow
↚	\nleftarrow	⇎	\nLeftrightarrow	⇏	\nRightarrow

Table 77: mathabx Harpoons

⥋	\barleftharpoon	↼	\leftharpoonup	⥤	\rightleftharpoons
⥊	\barrightharpoon	⇚	\leftleftharpoons	⇉	\rightrightharpoons
⥥	\downdownharpoons	↽	\leftrightharpoon	⥮	\updownharpoons
⥗	\downharpoonleft	⇋	\leftrightharpoons	↿	\upharpoonleft
⥖	\downharpoonright	⥌	\rightbarharpoon	↾	\upharpoonright
⥯	\downupharpoons	⇁	\rightharpoondown	⥣	\upupharpoons
⥎	\leftbarharpoon	⇀	\rightharpoonup		
↼	\leftharpoondown	⇀	\rightleftharpoon		

Table 78: chemarrow Arrows

→	\chemarrow

Table 79: ulsy Contradiction Symbols

↯	\blitza	↯	\blitzb	↯	\blitzc	↯	\blitzd	↯	\blitze

Table 80: Extension Characters

–	\relbar	=	\Relbar

Table 81: stmaryrd Extension Characters

⫽	\Arrownot	⫯	\Mapsfromchar	⫰	\Mapstochar
⫻	\arrownot	⫰	\mapsfromchar		

Table 82: txfonts/pxfonts Extension Characters

⫲	\Mappedfromchar	⫳	\Mmappedfromchar	⫴	\Mmapstochar
⫲	\mappedfromchar	⫳	\mmappedfromchar	⫴	\mmapstochar

Table 83: mathabx Extension Characters

⫰	\mapsfromchar	⫰	\mapstochar
⫯	\Mapsfromchar	⫯	\Mapstochar

Table 84: Log-like Symbols

`\arccos`	`\cos`	`\csc`	`\exp`	`\ker`	`\limsup`	`\min`	`\sinh`
`\arcsin`	`\cosh`	`\deg`	`\gcd`	`\lg`	`\ln`	`\Pr`	`\sup`
`\arctan`	`\cot`	`\det`	`\hom`	`\lim`	`\log`	`\sec`	`\tan`
`\arg`	`\coth`	`\dim`	`\inf`	`\liminf`	`\max`	`\sin`	`\tanh`

Calling the above "symbols" may be a bit misleading.[1] Each log-like symbol merely produces the eponymous textual equivalent, but with proper surrounding spacing. See Section 7.3 for more information about log-like symbols. As `\bmod` and `\pmod` are arguably not symbols we refer the reader to the Short Math Guide for LaTeX [Dow00] for samples.

Table 85: $\mathcal{AMS}$ Log-like Symbols

inj lim	`\injlim`	$\varinjlim$	`\varinjlim`	$\varlimsup$	`\varlimsup`
proj lim	`\projlim`	$\varliminf$	`\varliminf`	$\varprojlim$	`\varprojlim`

Load the amsmath package to get these symbols. See Section 7.3 for some additional comments regarding log-like symbols. As `\mod` and `\pod` are arguably not symbols we refer the reader to the Short Math Guide for LaTeX [Dow00] for samples.

Table 86: Greek Letters

α	`\alpha`	θ	`\theta`	o	`o`	τ	`\tau`
β	`\beta`	ϑ	`\vartheta`	π	`\pi`	υ	`\upsilon`
γ	`\gamma`	ι	`\iota`	ϖ	`\varpi`	ϕ	`\phi`
δ	`\delta`	κ	`\kappa`	ρ	`\rho`	φ	`\varphi`
ϵ	`\epsilon`	λ	`\lambda`	ϱ	`\varrho`	χ	`\chi`
ε	`\varepsilon`	μ	`\mu`	σ	`\sigma`	ψ	`\psi`
ζ	`\zeta`	ν	`\nu`	ς	`\varsigma`	ω	`\omega`
η	`\eta`	ξ	`\xi`				
Γ	`\Gamma`	Λ	`\Lambda`	Σ	`\Sigma`	Ψ	`\Psi`
Δ	`\Delta`	Ξ	`\Xi`	Υ	`\Upsilon`	Ω	`\Omega`
Θ	`\Theta`	Π	`\Pi`	Φ	`\Phi`		

The remaining Greek majuscules can be produced with ordinary Latin letters. The symbol "M", for instance, is used for both an uppercase "m" and an uppercase "μ". See Section 7.4 for examples of how to produce bold Greek letters.

Table 87: $\mathcal{AMS}$ Greek Letters

$\digamma$	`\digamma`	$\varkappa$	`\varkappa`

[1] Michael J. Downes prefers the more general term, "atomic math objects".

Table 88: txfonts/pxfonts Upright Greek Letters

α	\alphaup	θ	\thetaup	π	\piup	ϕ	\phiup
β	\betaup	ϑ	\varthetaup	ϖ	\varpiup	φ	\varphiup
γ	\gammaup	ι	\iotaup	ρ	\rhoup	χ	\chiup
δ	\deltaup	κ	\kappaup	ϱ	\varrhoup	ψ	\psiup
ϵ	\epsilonup	λ	\lambdaup	σ	\sigmaup	ω	\omegaup
ε	\varepsilonup	μ	\muup	ς	\varsigmaup		
ζ	\zetaup	ν	\nuup	τ	\tauup		
η	\etaup	ξ	\xiup	υ	\upsilonup		

Table 89: upgreek Upright Greek Letters

$\upalpha$	\upalpha	$\uptheta$	\uptheta	$\uppi$	\uppi	$\upphi$	\upphi
$\upbeta$	\upbeta	$\upvartheta$	\upvartheta	$\upvarpi$	\upvarpi	$\upvarphi$	\upvarphi
$\upgamma$	\upgamma	$\upiota$	\upiota	$\uprho$	\uprho	$\upchi$	\upchi
$\updelta$	\updelta	$\upkappa$	\upkappa	$\upvarrho$	\upvarrho	$\uppsi$	\uppsi
$\upepsilon$	\upepsilon	$\uplambda$	\uplambda	$\upsigma$	\upsigma	$\upomega$	\upomega
$\upvarepsilon$	\upvarepsilon	$\upmu$	\upmu	$\upvarsigma$	\upvarsigma		
$\upzeta$	\upzeta	$\upnu$	\upnu	$\uptau$	\uptau		
$\upeta$	\upeta	$\upxi$	\upxi	$\upupsilon$	\upupsilon		
$\Upgamma$	\Upgamma	$\Uplambda$	\Uplambda	$\Upsigma$	\Upsigma	$\Uppsi$	\Uppsi
$\Updelta$	\Updelta	$\Upxi$	\Upxi	$\Upupsilon$	\Upupsilon	$\Upomega$	\Upomega
$\Uptheta$	\Uptheta	$\Uppi$	\Uppi	$\Upphi$	\Upphi		

upgreek utilizes upright Greek characters from either the PostScript Symbol font (depicted above) or Euler Roman. As a result, the glyphs may appear slightly different from the above. Contrast, for example, "$\Gamma\Delta\Theta\alpha\beta\gamma$" (Symbol) with "$\Gamma\Delta\Theta\alpha\beta\gamma$" (Euler).

Table 90: txfonts/pxfonts Variant Latin Letters

g	\varg	v	\varv	w	\varw	y	\vary

Pass the varg option to txfonts/pxfonts to replace g, v, w, and y with g, v, w, and y in every mathematical expression in your document.

Table 91: $\mathcal{AMS}$ Hebrew Letters

$\beth$	\beth	$\gimel$	\gimel	$\daleth$	\daleth

\aleph appears in Table 125 on page 38.

Table 92: Letter-like Symbols

$\bot$	\bot	$\forall$	\forall	$\imath$	\imath	$\ni$	\ni	$\top$	\top
ℓ	\ell	$\hbar$	\hbar	$\in$	\in	∂	\partial	$\wp$	\wp
$\exists$	\exists	$\Im$	\Im	$\jmath$	\jmath	$\Re$	\Re		

TABLE 93: $\mathcal{AMS}$ Letter-like Symbols

$\Bbbk$	\Bbbk	$\complement$	\complement	$\hbar$	\hbar
®	\circledR	$\Finv$	\Finv	$\hslash$	\hslash
Ⓢ	\circledS	$\Game$	\Game	$\nexists$	\nexists

TABLE 94: txfonts/pxfonts Letter-like Symbols

¢	\mathcent	£	\mathsterling	$\notin$	\notin	$\notni$	\notni

TABLE 95: mathabx Letter-like Symbols

$\bar{\in}$	\barin	$\in$	\in	$\nottop$	\nottop	$\varnotin$	\varnotin
$\complement$	\complement	$\nexists$	\nexists	$\ni$	\owns	$\varnotowner$	\varnotowner
$\exists$	\exists	$\notbot$	\notbot	$\supseteq$	\ownsbar		
$\Finv$	\Finv	$\notin$	\notin	∂	\partial		
$\Game$	\Game	$\notni$	\notowner	$\partialslash$	\partialslash		

TABLE 96: trfsigns Letter-like Symbols

e	\e	j	\im

TABLE 97: $\mathcal{AMS}$ Delimiters

⌜	\ulcorner	⌝	\urcorner
⌞	\llcorner	⌟	\lrcorner

TABLE 98: stmaryrd Delimiters

⦉	\Lbag	⦊	\Rbag	⦉	\lbag	⦊	\rbag
⌈⌈	\llceil	⌉⌉	\rrceil	⌊⌊	\llfloor	⌋⌋	\rrfloor
⦇	\llparenthesis	⦈	\rrparenthesis				

TABLE 99: mathabx Delimiters

⌜⌟	\lcorners	⌜⌟	\rcorners
⌜	\ulcorner	⌝	\urcorner
⌞	\llcorner	⌟	\lrcorner

TABLE 100: nath Delimiters

⌞	\niv	⌟	\vin

TABLE 101: Variable-sized Delimiters

↓	↓	\downarrow	⇓	⇓	\Downarrow	[	[	[	]	]	]
⟨	⟨	\langle	⟩	⟩	\rangle	|	|	|*	‖	‖	\\|
⌈	⌈	\lceil	⌉	⌉	\rceil	↑	↑	\uparrow	⇑	⇑	\Uparrow
⌊	⌊	\lfloor	⌋	⌋	\rfloor	↕	↕	\updownarrow	⇕	⇕	\Updownarrow
(	(	(	)	)	)	{	{	\{	}	}	\}
/	/	/	\	\	\backslash						

When used with `\left` and `\right`, these symbols expand to the height of the enclosed math expression. Note that `\vert` is a synonym for `|`, and `\Vert` is a synonym for `\|`.

* ε-TEX provides a `\middle` analogue to `\left` and `\right` that can be used to make an internal "|" (often used to indicate "evaluated at") expand to the height of the surrounding `\left` and `\right` symbols. A similar effect can be achieved in conventional LATEX using the braket package.

TABLE 102: Large, Variable-sized Delimiters

∫	∫	\lmoustache	)	)	\rmoustache	(	(	\lgroup	)	)	\rgroup
|	|	\arrowvert	‖	‖	\Arrowvert	▏	▏	\bracevert			

These symbols *must* be used with `\left` and `\right`. The mathabx package, however, redefines `\lgroup` and `\rgroup` so that those symbols can work without `\left` and `\right`.

TABLE 103: Variable-sized stmaryrd Delimiters

⟦	⟦	\llbracket	⟧	⟧	\rrbracket

TABLE 104: mathabx Variable-sized Delimiters

⟦	⟦	\lbbbrack	⟧	⟧	\rbbbrack
❨	❨	\lfilet	❩	❩	\rfilet
|	|	\thickvert	⦀	⦀	\vvvert

Table 105: nath Variable-sized Delimiters (Double)

⟪	⟪	\lAngle	⟫	⟫	\rAngle
⟦	⟦	\lBrack	⟧	⟧	\rBrack
⟮	⟮	\lCeil	⟯	⟯	\rCeil
⟬	⟬	\lFloor	⟭	⟭	\rFloor
‖	‖	\lVert*	‖	‖	\rVert*

* nath redefines all of the above to include implicit \left and \right commands. Hence, separate \lVert and \rVert commands are needed to disambiguate whether "|" is a left or right delimiter.

All of the symbols in Table 105 can also be expressed using the \double macro. See the nath documentation for examples and additional information.

Table 106: nath Variable-sized Delimiters (Triple)

⟪⟪	⟪⟪	\triple<	⟫⟫	⟫⟫	\triple>
⟦⟦	⟦⟦	\triple[	⟧⟧	⟧⟧	\triple]
⦀	⦀	\ltriple|*	⦀	⦀	\rtriple|*

* Similar to \lVert and \rVert in Table 105, \ltriple and \rtriple must be used instead of \triple to disambiguate whether "|" is a left or right delimiter.

Note that \triple—and the corresponding \double—is actually a macro that takes a delimiter as an argument.

Table 107: textcomp Text-mode Delimiters

⟨	\textlangle	⟩	\textrangle
⟦	\textlbrackdbl	⟧	\textrbrackdbl
{	\textlquill	}	\textrquill

Table 108: Math-mode Accents

$\acute{a}$	`\acute{a}`	$\check{a}$	`\check{a}`	$\grave{a}$	`\grave{a}`	$\tilde{a}$	`\tilde{a}`
$\bar{a}$	`\bar{a}`	$\ddot{a}$	`\ddot{a}`	$\hat{a}$	`\hat{a}`	$\vec{a}$	`\vec{a}`
$\breve{a}$	`\breve{a}`	$\dot{a}$	`\dot{a}`	$\mathring{a}$	`\mathring{a}`		

Also note the existence of `\imath` and `\jmath`, which produce dotless versions of "$\imath$" and "$\jmath$". (See Table 125 on page 38.) These are useful when the accent is supposed to replace the dot. For example, "`\hat{\imath}`" produces a correct "$\hat{\imath}$", while "`\hat{i}`" would yield the rather odd-looking "$\hat{i}$".

Table 109: $\mathcal{AMS}$ Math-mode Accents

$\ddot{a}$	`\dddot{a}`	$\ddddot{a}$	`\ddddot{a}`

These accents are also provided by the mathabx package.

Table 110: yhmath Math-mode Accents

$\mathring{a}$	`\ring{a}`

This symbol is largely obsolete, as standard LaTeX 2_ε has supported `\mathring` since June, 1998 [LaT98].

Table 111: trfsigns Math-mode Accents

$\underset{a}{\vdash\!\!-}$	`\dft{a}`	$\underset{a}{-\!\!\dashv}$	`\DFT{a}`

The above are a sort of "reverse accent" in that the argument text serves as a subscript to the transform line.

TABLE 112: Extensible Accents

$\widetilde{abc}$	`\widetilde{abc}`*	$\widehat{abc}$	`\widehat{abc}`*
$\overleftarrow{abc}$	`\overleftarrow{abc}`†	$\overrightarrow{abc}$	`\overrightarrow{abc}`†
$\overline{abc}$	`\overline{abc}`	$\underline{abc}$	`\underline{abc}`
$\overbrace{abc}$	`\overbrace{abc}`	$\underbrace{abc}$	`\underbrace{abc}`
$\sqrt{abc}$	`\sqrt{abc}`‡		

As demonstrated in a 1997 TUGboat article about typesetting long-division problems [Gib97], an extensible long-division sign ("$)abc$") can be faked by putting a "`\big)`" in a `tabular` environment with an `\hline` or `\cline` in the preceding row. The article also presents a piece of code that automatically solves and typesets—by putting an `\overline` atop "`\big)`" and the desired text—long-division problems. See also the polynom package, which automatically solves and typesets polynomial-division problems in a similar manner.

* Made more extensible by the yhmath package.

† If you're looking for an extensible *diagonal* line or arrow to be used for canceling or reducing mathematical subexpressions (e.g., "$\cancel{x+=x}$" or "$\cancel{3+2}^{5}$") then consider using the cancel package.

‡ With an optional argument, `\sqrt` typesets nth roots. For example, "`\sqrt[3]{abc}`" produces "$\sqrt[3]{abc}$" and "`\sqrt[n]{abc}`" produces "$\sqrt[n]{abc}$".

TABLE 113: overrightarrow Extensible Accents

$\overRightarrow{abc}$	`\Overrightarrow{abc}`

TABLE 114: yhmath Extensible Accents

$\wideparen{abc}$	`\wideparen{abc}`	$\widetriangle{abc}$	`\widetriangle{abc}`
$\widering{abc}$	`\widering{abc}`		

TABLE 115: $\mathcal{AMS}$ Extensible Accents

$\overleftrightarrow{abc}$	`\overleftrightarrow{abc}`	$\underleftrightarrow{abc}$	`\underleftrightarrow{abc}`
$\underleftarrow{abc}$	`\underleftarrow{abc}`	$\underrightarrow{abc}$	`\underrightarrow{abc}`

The following are a sort of "reverse accent" in that the argument text serves as a superscript to the arrow. In addition, the optional first argument (not shown) serves as a subscript to the arrow. See the Short Math Guide for LaTeX [Dow00] for further examples.

$\xleftarrow{abc}$	`\xleftarrow{abc}`	$\xrightarrow{abc}$	`\xrightarrow{abc}`

Table 116: chemarr Extensible Accents

$\overset{abc}{\rightleftharpoons}$ `\xrightleftharpoons{abc}`

`\xrightleftharpoons` is a sort of "reverse accent" in that the argument text serves as a superscript to the arrows. In addition, the optional first argument (not shown) serves as a subscript to the arrows.

Table 117: chemarrow Extensible Accents

$\xleftarrow[def]{abc}$ `\autoleftarrow{abc}{def}` $\xrightarrow[def]{abc}$ `\autorightarrow{abc}{def}`

$\xleftrightharpoons[def]{abc}$ `\autoleftrightharpoons{abc}{def}` $\xrightleftharpoons[def]{abc}$ `\autorightleftharpoons{abc}{def}`

These symbols are all "reverse accents" in that the two arguments serve, respectively, as a superscript and a subscript to the arrows.

In addition to the symbols shown above, chemarrow also provides `\larrowfill`, `\rarrowfill`, `\leftrightharpoonsfill`, and `\rightleftharpoonsfill` macros. Each of these takes a length argument and produces an arrow of the specified length.

Table 118: mathabx Extensible Accents

$\overbrace{abc}$	`\overbrace{abc}`	$\overline{abc}$	`\widebar{abc}`
$\overgroup{abc}$	`\overgroup{abc}`	$\widecheck{abc}$	`\widecheck{abc}`
$\underbrace{abc}$	`\underbrace{abc}`	$\wideparen{abc}$	`\wideparen{abc}`
$\undergroup{abc}$	`\undergroup{abc}`	$\mathring{abc}$	`\widering{abc}`
$\overrightarrow{abc}$	`\widearrow{abc}`		

The braces shown for `\overbrace` and `\underbrace` appear in their minimum size. They can expand arbitrarily wide, however.

Table 119: esvect Extensible Accents

$\overrightarrow{abc}$	`\vv{abc}` with package option `a`
$\overrightarrow{abc}$	`\vv{abc}` with package option `b`
$\overrightarrow{abc}$	`\vv{abc}` with package option `c`
$\overrightarrow{abc}$	`\vv{abc}` with package option `d`
$\overrightarrow{abc}$	`\vv{abc}` with package option `e`
$\overrightarrow{abc}$	`\vv{abc}` with package option `f`
$\overrightarrow{abc}$	`\vv{abc}` with package option `g`
$\overrightarrow{abc}$	`\vv{abc}` with package option `h`

esvect also defines a `\vv*` macro which is used to typeset arrows over vector variables with subscripts. See the esvect documentation for more information.

Table 120: undertilde Extensible Accents

$\underset{\sim}{abc}$	`\utilde{abc}`

Because `\utilde` is based on `\widetilde` it is also made more extensible by the yhmath package.

Table 121: Dots

$\cdotp$	`\cdotp`	:	`\colon`*	.	`\ldotp`	$\vdots$	`\vdots`†
$\cdots$	`\cdots`	$\ddots$	`\ddots`†	...	`\ldots`		

* While ":" is valid in math mode, `\colon` uses different surrounding spacing. See Section 7.3 and the Short Math Guide for LaTeX [Dow00] for more information on math-mode spacing.

† The mathdots package redefines `\ddots` and `\vdots` to make them scale properly with font size. (They normally scale horizontally but not vertically.) `\fixedddots` and `\fixedvdots` provide the original, fixed-height functionality of LaTeX 2_ε's `\ddots` and `\vdots` macros.

Table 122: $\mathcal{AMS}$ Dots

$\cdots$	`\dotsb`	$\cdots$	`\dotsi`	...	`\dotso`
...	`\dotsc`	$\cdots$	`\dotsm`		

The $\mathcal{AMS}$ dot symbols are named according to their intended usage: `\dotsb` between pairs of binary operators/relations, `\dotsc` between pairs of commas, `\dotsi` between pairs of integrals, `\dotsm` between pairs of multiplication signs, and `\dotso` between other symbol pairs.

TABLE 123: mathdots Dots

⋰	\iddots

TABLE 124: yhmath Dots

⋰	\adots

TABLE 125: Miscellaneous LaTeX 2_ε Symbols

ℵ	\aleph	◇	\Diamond*	∞	\infty	′	\prime
∠	\angle	♢	\diamondsuit	℧	\mho*	♯	\sharp
\	\backslash	∅	\emptyset‡	∇	\nabla	♠	\spadesuit
□	\Box*,†	♭	\flat	♮	\natural	√	\surd
♣	\clubsuit	♡	\heartsuit	¬	\neg	△	\triangle

* Not predefined in LaTeX 2_ε. Use one of the packages latexsym, amsfonts, amssymb, txfonts, pxfonts, or wasysym.

† To use \Box—or any other symbol—as an end-of-proof (Q.E.D.) marker, consider using the ntheorem package, which properly juxtaposes a symbol with the end of the proof text.

‡ Many people prefer the look of $\mathcal{AMS}$'s \varnothing (Table 126) to that of LaTeX's \emptyset.

TABLE 126: Miscellaneous $\mathcal{AMS}$ Symbols

∠	\angle	▼	\blacktriangledown	℧	\mho
‵	\backprime	╲	\diagdown	◁	\sphericalangle
★	\bigstar	╱	\diagup	□	\square
⧫	\blacklozenge	ð	\eth	▽	\triangledown
■	\blacksquare	◊	\lozenge	∅	\varnothing
▲	\blacktriangle	∡	\measuredangle	△	\vartriangle

TABLE 127: Miscellaneous wasysym Symbols

□	\Box	℧	\mho*	∴	\wasytherefore
◇	\Diamond	⍀	\varangle		

* wasysym also defines an \agemO symbol, which is the same glyph as \mho but is intended for use in text mode.

TABLE 128: Miscellaneous txfonts/pxfonts Symbols

◆	\Diamondblack	ƛ	\lambdaslash	♥	\varheartsuit
⟐	\Diamonddot	♧	\varclubsuit	♤	\varspadesuit
ƛ	\lambdabar	♦	\vardiamondsuit		

Table 129: Miscellaneous mathabx Symbols

○	\degree	⫽⫽	\fourth	∡	\measuredangle	⫽	\second
⧹	\diagdown	#	\hash	⫛	\pitchfork	∢	\sphericalangle
⧸	\diagup	∞	\infty	∝	\propto	⫽	\third
∅	\diameter	⋉	\leftthreetimes	⋌	\rightthreetimes	#	\varhash

Table 130: Miscellaneous textcomp Text-mode Math Symbols

°	\textdegree*	½	\textonehalf†	¾	\textthreequarters†
÷	\textdiv	¼	\textonequarter†	³	\textthreesuperior
/	\textfractionsolidus	¹	\textonesuperior	×	\texttimes
¬	\textlnot	±	\textpm	²	\texttwosuperior
−	\textminus	√	\textsurd		

* If you prefer a larger degree symbol you might consider defining one as "\ensuremath{^\circ}" ("°").

† nicefrac (part of the units package) can be used to construct vulgar fractions like "1/2", "1/4", "3/4", and even "c/o".

Table 131: mathcomp Math Symbols

℃	\tccentigrade	Ω	\tcohm	‰	\tcperthousand
µ	\tcmu	‱	\tcpertenthousand		

Table 132: gensymb Symbols Defined to Work in Both Math and Text Mode

℃	\celsius	µ	\micro	‰	\perthousand
°	\degree	Ω	\ohm		

Table 133: mathabx Mayan Digits

𝋠	\maya{0}	𝋡	\maya{2}	𝋢	\maya{4}
𝋟	\maya{1}	𝋢	\maya{3}	𝋣	\maya{5}

Table 134: marvosym Math Symbols

0	\MVZero	2	\MVTwo	4	\MVFour	6	\MVSix	8	\MVEight
1	\MVOne	3	\MVThree	5	\MVFive	7	\MVSeven	9	\MVNine

∢	\Anglesign	·	\Squaredot	⃗	\Vectorarrowhigh
≙	\Corresponds	⃗	\Vectorarrow		

Table 135: Math Alphabets

		Required package
ABCdef123	`\mathrm{ABCdef123}`	*none*
$ABCdef123$	`\mathit{ABCdef123}`	*none*
$ABCdef123$	`\mathnormal{ABCdef123}`	*none*
$\mathcal{ABC}$	`\mathcal{ABC}`	*none*
$\mathscr{ABC}$	`\mathscr{ABC}`	mathrsfs
or	`\mathcal{ABC}`	calrsfs
$\mathcal{ABC}$	`\mathcal{ABC}`	euscript with the mathcal option
or	`\mathscr{ABC}`	euscript with the mathscr option
$\mathpzc{ABCdef123}$	`\mathpzc{ABCdef123}`	*none*; manually defined*
$\mathbb{ABC}$	`\mathbb{ABC}`	amsfonts,§ amssymb, txfonts, or pxfonts
$\mathbf{\mathbb{ABC}}$	`\varmathbb{ABC}`	txfonts or pxfonts
$\mathbb{ABCdef123}$	`\mathbb{ABCdef123}`	bbold or mathbbol†
$\mathbb{ABCdef123}$	`\mathbb{ABCdef123}`	mbboard†
$\mathbb{ABCdef12}$	`\mathbbm{ABCdef12}`	bbm
$\mathbb{ABCdef12}$	`\mathbbmss{ABCdef12}`	bbm
$\mathbb{ABCdef12}$	`\mathbbmtt{ABCdef12}`	bbm
$\mathbb{ABC1}$	`\mathds{ABC1}`	dsfont
$\mathbf{\mathbb{ABC1}}$	`\mathds{ABC1}`	dsfont with the sans option
$\mathfrak{ABCdef123}$	`\mathfrak{ABCdef123}`	eufrak
$\mathfrak{ABCdef123}$	`\textfrak{ABCdef123}`	yfonts‡
$\mathfrak{ABCdef123}$	`\textswab{ABCdef123}`	yfonts‡
$\mathfrak{ABCdef123}$	`\textgoth{ABCdef123}`	yfonts‡

* Put "`\DeclareMathAlphabet{\mathpzc}{OT1}{pzc}{m}{it}`" in your document's preamble to make `\mathpzc` typeset its argument in Zapf Chancery.

† The mathbbol package defines some additional blackboard bold characters: parentheses, square brackets, angle brackets, and—if the bbgreekl option is passed to matbbol—Greek letters. For instance, "$\leqslant[\![(\alpha\beta\delta)]\!]\geqslant$" is produced by "`\mathbb{\Langle\Lbrack\Lparen\bbalpha\bbbeta\bbgamma\Rparen\Rbrack\Rangle}`".

mbboard extends the blackboard bold symbol set significantly further. It supports not only the Greek alphabet—including "Greek-like" symbols such as `\bbnabla` ("$\mathbb{\nabla}$")—but also *all* punctuation marks, various currency symbols such as `\bbdollar` ("$") and `\bbeuro` ("€"), and the Hebrew alphabet (e.g., "`\bbfinalnun\bbyod\bbqof\bbpe`" → "$\mathbb{\text{ן}יקפ}$").

‡ As their `\text...` names imply, the fonts provided by the yfonts package are actually text fonts. They are included in Table 135 because they are frequently used in a mathematical context.

§ An older (i.e., prior to 1991) version of the $\mathcal{AMS}$'s fonts rendered $\mathbb{C}$, $\mathbb{N}$, $\mathbb{R}$, $\mathbb{S}$, and $\mathbb{Z}$ as $\mathbb{C}$, $\mathbb{N}$, $\mathbb{R}$, $\mathbb{S}$, and $\mathbb{Z}$. As some people prefer the older glyphs—much to the $\mathcal{AMS}$'s surprise—and because those glyphs fail to build under modern versions of METAFONT, Berthold Horn uploaded PostScript fonts for the older blackboard-bold glyphs to CTAN, to the `fonts/msym10` directory. As of this writing, however, there are no LaTeX 2_ε packages for utilizing the now-obsolete glyphs.

4 Science and technology symbols

This section lists symbols that are employed in various branches of science and engineering (and, because we were extremely liberal in our classification, astrology, too).

Table 136: wasysym Electrical and Physical Symbols

∼	\AC	≋	\VHF	∿∿∿	\photon	≈	\HF	∿∿∿	\gluon

Table 137: ifsym Pulse Diagram Symbols

⎿	\FallingEdge	⎾⏌	\LongPulseLow	⎾⏌	\PulseLow	⏍	\ShortPulseHigh
⏍⏌	\LongPulseHigh	⏍⏌	\PulseHigh	⏍	\RaisingEdge	⎾	\ShortPulseLow

In addition, within \textifsym{...}, the following codes are valid:

_	l	¯	m	‾	h	‗	d	〈	<	〉	>
__	L	¯¯	M	‾‾	H	‗‗	D	〈	<<	〉	>>

This enables one to write "\textifsym{mm<DDD>mm}" to get "⟵⬭⟶" or "\textifsym{L|H|L|H|L}" to get "⎽⎍⎽⎍⎽". See also the timing package, which provides a wide variety of pulse-diagram symbols within an environment designed specifically for typesetting pulse diagrams.

Finally, \textifsym supports the display of segmented digits, as would appear on an LCD: "\textifsym{-123.456}" produces "- 123.456". "\textifsym{b}" outputs a blank with the same width as an "8".

Table 138: ar Aspect Ratio Symbol

$\mathcal{R}$	\AR

Table 139: textcomp Text-mode Science and Engineering Symbols

℃	\textcelsius	℧	\textmho	µ	\textmu	Ω	\textohm

Table 140: wasysym Astronomical Symbols

☊	\ascnode	♃	\jupiter	●	\newmoon	♀	\venus
☉	\astrosun	☾	\leftmoon	♇	\pluto	♈	\vernal
☋	\descnode	♂	\mars	☽	\rightmoon		
♁	\earth	☿	\mercury	♄	\saturn		
○	\fullmoon	♆	\neptune	♁	\uranus		

Table 141: marvosym Astronomical Symbols

☿	\Mercury	♂	\Mars	♅	\Uranus	☉	\Sun
♀	\Venus	♃	\Jupiter	♆	\Neptune	☽	\Moon
♁	\Earth	♄	\Saturn	♇	\Pluto		

Table 142: mathabx Astronomical Symbols

☿	\Mercury	⊕	\Earth	♃	\Jupiter	♅	\Uranus	♇	\Pluto
♀	\Venus	♂	\Mars	♄	\Saturn	♆	\Neptune		
○	\fullmoon	☾	\leftmoon	●	\newmoon	☽	\rightmoon		
⊙	\Sun	♁	\varEarth						

mathabx also defines \girl as an alias for \Venus, \boy as an alias for \Mars, and \Moon as an alias for \leftmoon.

Table 143: wasysym Astrological Symbols

♈	\aries	♋	\cancer	♎	\libra	♑	\capricornus
♉	\taurus	♌	\leo	♏	\scorpio	♒	\aquarius
♊	\gemini	♍	\virgo	♐	\sagittarius	♓	\pisces

☌	\conjunction	☍	\opposition

Table 144: marvosym Astrological Symbols

♈	\Aries	♋	\Cancer	♎	\Libra	♑	\Capricorn
♉	\Taurus	♌	\Leo	♏	\Scorpio	♒	\Aquarius
♊	\Gemini	♍	\Virgo	♐	\Sagittarius	♓	\Pisces

Note that \Aries...\Pisces can also be specified with \Zodiac{1}... \Zodiac{12}.

Table 145: mathabx Astrological Symbols

♈	\Aries	♉	\Taurus	♊	\Gemini

Table 146: wasysym APL Symbols

⍞	\APLbox	⍠	\APLinv	⋆	\APLstar
⍝	\APLcomment	⍇	\APLleftarrowbox	⍙	\APLup
⍛	\APLdown	⍟	\APLlog	⍐	\APLuparrowbox
⍗	\APLdownarrowbox	¯	\APLminus	⍀	\notbackslash
⍞	\APLinput	⍈	\APLrightarrowbox	⌿	\notslash

Table 147: wasysym APL Modifiers

∘	\APLcirc{}	∼	\APLnot{}	|	\APLvert{}

TABLE 148: marvosym Computer Hardware Symbols

	\ComputerMouse		\ParallelPort		\SerialInterface
	\Keyboard		\Printer		\SerialPort

TABLE 149: ascii Control Characters (IBM)

☺	\SOH	•	\BEL	♪	\CR	‼	\DCc	↓	\EM	▼	\US
☻	\STX	◘	\BS	♬	\SO	¶	\DCd	→	\SUB	¦	\splitvert
♥	\ETX	○	\HT	☼	\SI	§	\NAK	←	\ESC	⌂	\DEL
♦	\EOT	◙	\LF	►	\DLE	▬	\SYN	└	\FS		
♣	\ENQ	♂	\VT	◄	\DCa	↨	\ETB	↔	\GS		
♠	\ACK	♀	\FF	↕	\DCb	↑	\CAN	▲	\RS		

SOH, STX, ETX, ..., US are the names of ASCII characters 1–31. DEL is the name of ASCII character 127. \splitvert doesn't correspond to a control character but is merely the "|" character shown IBM style.

These characters must be entered with the ascii font in effect, for example, "{\ascii\STX}". See the ascii package documentation for more information.

TABLE 150: marvosym Communication Symbols

	\Email	FAX	\fax		\Faxmachine		\Lightning		\Pickup
	\Emailct	FAX	\FAX	✉	\Letter		\Mobilefone	☎	\Telefon

TABLE 151: marvosym Engineering Symbols

	\Beam	↓	\Force	●	\Octosteel	I	\RoundedTTsteel
	\Bearing	⬢	\Hexasteel	◘	\Rectpipe	◘	\Squarepipe
O	\Circpipe	↺	\Lefttorque	■	\Rectsteel	■	\Squaresteel
●	\Circsteel	⇈	\Lineload	↻	\Righttorque	T	\Tsteel
	\Fixedbearing		\Loosebearing	T	\RoundedLsteel*	I	\TTsteel
▬	\Flatsteel	L	\Lsteel	L	\RoundedTsteel*		

* \RoundedLsteel and \RoundedTsteel seem to be swapped, at least in the 2000/05/01 version of marvosym.

TABLE 152: wasysym Biological Symbols

♀	\female	♂	\male

TABLE 153: marvosym Biological Symbols

♀	\Female	⚥	\FemaleMale	⚷	\MALE	○	\Neutral
⚲	\FEMALE	⚥	\Hermaphrodite	♂	\Male		
⚢	\FemaleFemale	⚥	\HERMAPHRODITE	⚣	\MaleMale		

TABLE 154: marvosym Safety-related Symbols

☣	\Biohazard	CE	\CEsign	Ex	\Explosionsafe	☢	\Radioactivity
BSE	\BSEfree	⚠	\Estatically	✷—	\Laserbeam	STOP	\Stopsign

5 Dingbats

Dingbats are symbols such as stars, arrows, and geometric shapes. They are commonly used as bullets in itemized lists or, more generally, as a means to draw attention to the text that follows.

The pifont dingbat package warrants special mention. Among other capabilities, pifont provides a LaTeX interface to the Zapf Dingbats font (one of the standard 35 PostScript fonts). However, rather than name each of the dingbats individually, pifont merely provides a single `\ding` command, which outputs the character that lies at a given position in the font. The consequence is that the pifont symbols can't be listed by name in this document's index, so be mindful of that fact when searching for a particular symbol.

Table 155: bbding Arrows

➥	\ArrowBoldDownRight	➧	\ArrowBoldRightShort	➦	\ArrowBoldUpRight
➲	\ArrowBoldRightCircled	➠	\ArrowBoldRightStrobe		

Table 156: pifont Arrows

➔	\ding{212}	→	\ding{221}	➦	\ding{230}	⇨	\ding{239}	➶	\ding{249}
→	\ding{213}	➙	\ding{222}	➧	\ding{231}	⇨	\ding{241}	➴	\ding{250}
↔	\ding{214}	➛	\ding{223}	➨	\ding{232}	➲	\ding{242}	➵	\ding{251}
↕	\ding{215}	➠	\ding{224}	⇨	\ding{233}	➳	\ding{243}	➸	\ding{252}
↘	\ding{216}	➡	\ding{225}	⇨	\ding{234}	➷	\ding{244}	➹	\ding{253}
➙	\ding{217}	➢	\ding{226}	⇨	\ding{235}	➸	\ding{245}	⇛	\ding{254}
➚	\ding{218}	➢	\ding{227}	⇨	\ding{236}	➶	\ding{246}		
➛	\ding{219}	➤	\ding{228}	⇨	\ding{237}	➷	\ding{247}		
➔	\ding{220}	➥	\ding{229}	⇨	\ding{238}	➸	\ding{248}		

Table 157: marvosym Scissors

✁	\Cutleft	✂	\Cutright	✁	\Leftscissors
---	\Cutline	---	\Kutline	✂	\Rightscissors

Table 158: bbding Scissors

✃	\ScissorHollowLeft	✃	\ScissorLeftBrokenTop
✄	\ScissorHollowRight	✂	\ScissorRight
✁	\ScissorLeft	✂	\ScissorRightBrokenBottom
✃	\ScissorLeftBrokenBottom	✂	\ScissorRightBrokenTop

Table 159: pifont Scissors

✃	\ding{33}	✂	\ding{34}	✂	\ding{35}	✄	\ding{36}

Table 160: dingbat Pencils

	\largepencil		\smallpencil

TABLE 161: bbding Pencils and Nibs

✑	\NibLeft	✏	\PencilLeft	✎	\PencilRightDown
✒	\NibRight	✐	\PencilLeftDown	✐	\PencilRightUp
✒	\NibSolidLeft	✎	\PencilLeftUp		
✒	\NibSolidRight	✏	\PencilRight		

TABLE 162: pifont Pencils and Nibs

✎	\ding{46}	✏	\ding{47}	✐	\ding{48}	✑	\ding{49}	✒	\ding{50}

TABLE 163: dingbat Hands

☞	\leftpointright	☜	\rightpointleft	☞	\rightpointright
☟	\leftthumbsdown	☟	\rightthumbsdown		
☝	\leftthumbsup	☝	\rightthumbsup		

TABLE 164: bbding Hands

☚	\HandCuffLeft	☛	\HandCuffRightUp	✍	\HandPencilLeft
☚	\HandCuffLeftUp	☜	\HandLeft	☞	\HandRight
☛	\HandCuffRight	☜	\HandLeftUp	☞	\HandRightUp

TABLE 165: pifont Hands

☛	\ding{42}	☞	\ding{43}	✌	\ding{44}	✍	\ding{45}		

TABLE 166: bbding Crosses and Plusses

✝	\Cross	✞	\CrossOpenShadow	✙	\PlusOutline
✟	\CrossBoldOutline	✝	\CrossOutline	✛	\PlusThinCenterOpen
✣	\CrossClowerTips	✚	\Plus		
✠	\CrossMaltese	✜	\PlusCenterOpen		

TABLE 167: pifont Crosses and Plusses

✙	\ding{57}	✛	\ding{59}	✝	\ding{61}	✟	\ding{63}
✚	\ding{58}	✜	\ding{60}	✞	\ding{62}	✠	\ding{64}

TABLE 168: bbding Xs and Check Marks

✓	\Checkmark	╳	\XSolid	✗	\XSolidBrush
✔	\CheckmarkBold	✖	\XSolidBold		

Table 169: pifont Xs and Check Marks

✓	\ding{51}	✕	\ding{53}	✗	\ding{55}
✔	\ding{52}	✖	\ding{54}	✘	\ding{56}

Table 170: wasysym Xs and Check Marks

☑	\CheckedBox	□	\Square	☒	\XBox

Table 171: pifont Circled Numbers

①	\ding{172}	❶	\ding{182}	①	\ding{192}	❶	\ding{202}
②	\ding{173}	❷	\ding{183}	②	\ding{193}	❷	\ding{203}
③	\ding{174}	❸	\ding{184}	③	\ding{194}	❸	\ding{204}
④	\ding{175}	❹	\ding{185}	④	\ding{195}	❹	\ding{205}
⑤	\ding{176}	❺	\ding{186}	⑤	\ding{196}	❺	\ding{206}
⑥	\ding{177}	❻	\ding{187}	⑥	\ding{197}	❻	\ding{207}
⑦	\ding{178}	❼	\ding{188}	⑦	\ding{198}	❼	\ding{208}
⑧	\ding{179}	❽	\ding{189}	⑧	\ding{199}	❽	\ding{209}
⑨	\ding{180}	❾	\ding{190}	⑨	\ding{200}	❾	\ding{210}
⑩	\ding{181}	❿	\ding{191}	⑩	\ding{201}	❿	\ding{211}

pifont (part of the psnfss package) provides a `dingautolist` environment which resembles `enumerate` but uses circled numbers as bullets.[2] See the psnfss documentation for more information.

Table 172: wasysym Stars

✡	\davidsstar	⚹	\hexstar	⚹	\varhexstar

Table 173: bbding Stars, Flowers, and Similar Shapes

✻	\Asterisk	✿	\FiveFlowerPetal	✢	\JackStar
✱	\AsteriskBold	★	\FiveStar	✤	\JackStarBold
✲	\AsteriskCenterOpen	☆	\FiveStarCenterOpen	✾	\SixFlowerAlternate
✽	\AsteriskRoundedEnds	✭	\FiveStarConvex	❋	\SixFlowerAltPetal
✳	\AsteriskThin	✮	\FiveStarLines	✼	\SixFlowerOpenCenter
✴	\AsteriskThinCenterOpen	✯	\FiveStarOpen	❀	\SixFlowerPetalDotted
✡	\DavidStar	✪	\FiveStarOpenCircled	✻	\SixFlowerPetalRemoved
✸	\DavidStarSolid	✰	\FiveStarOpenDotted	❀	\SixFlowerRemovedOpenPetal
✷	\EightAsterisk	✰	\FiveStarOutline	✶	\SixStar
❁	\EightFlowerPetal	✭	\FiveStarOutlineHeavy	✺	\SixteenStarLight
✻	\EightFlowerPetalRemoved	✮	\FiveStarShadow	❄	\Snowflake
✶	\EightStar	✢	\FourAsterisk	❅	\SnowflakeChevron
✸	\EightStarBold	❀	\FourClowerOpen	❆	\SnowflakeChevronBold
✵	\EightStarConvex	✤	\FourClowerSolid	✻	\Sparkle
✶	\EightStarTaper	✦	\FourStar	✼	\SparkleBold
❀	\FiveFlowerOpen	✧	\FourStarOpen	✹	\TwelweStar

[2] In fact, `dingautolist` can use any set of consecutive Zapf Dingbats symbols.

Table 174: pifont Stars, Flowers, and Similar Shapes

✡	\ding{65}	✪	\ding{74}	✳	\ding{83}	✻	\ding{92}	❋	\ding{101}
✢	\ding{66}	✫	\ding{75}	✴	\ding{84}	✼	\ding{93}	❁	\ding{102}
✣	\ding{67}	✬	\ding{76}	✵	\ding{85}	✽	\ding{94}	✳	\ding{103}
✤	\ding{68}	✭	\ding{77}	✶	\ding{86}	✿	\ding{95}	✻	\ding{104}
✥	\ding{69}	✮	\ding{78}	✷	\ding{87}	❀	\ding{96}	✼	\ding{105}
✦	\ding{70}	✯	\ding{79}	✸	\ding{88}	❁	\ding{97}	✽	\ding{106}
✧	\ding{71}	✰	\ding{80}	✹	\ding{89}	✪	\ding{98}	❋	\ding{107}
★	\ding{72}	✱	\ding{81}	✺	\ding{90}	❋	\ding{99}		
☆	\ding{73}	✲	\ding{82}	✻	\ding{91}	❄	\ding{100}		

Table 175: wasysym Geometric Shapes

⬡ \hexagon ◯ \octagon ⬠ \pentagon ⬡ \varhexagon

Table 176: ifsym Geometric Shapes

◯	\BigCircle	▶	\FilledBigTriangleRight	○	\SmallCircle
╳	\BigCross	▲	\FilledBigTriangleUp	×	\SmallCross
◇	\BigDiamondshape	●	\FilledCircle	◇	\SmallDiamondshape
—	\BigHBar	◈	\FilledDiamondShadowA	–	\SmallHBar
⬙	\BigLowerDiamond	◈	\FilledDiamondShadowC	⬙	\SmallLowerDiamond
⬗	\BigRightDiamond	◆	\FilledDiamondshape	⬗	\SmallRightDiamond
□	\BigSquare	●	\FilledSmallCircle	□	\SmallSquare
▽	\BigTriangleDown	◆	\FilledSmallDiamondshape	▽	\SmallTriangleDown
◁	\BigTriangleLeft	■	\FilledSmallSquare	◁	\SmallTriangleLeft
▷	\BigTriangleRight	▼	\FilledSmallTriangleDown	▷	\SmallTriangleRight
△	\BigTriangleUp	◀	\FilledSmallTriangleLeft	△	\SmallTriangleUp
|	\BigVBar	▶	\FilledSmallTriangleRight	|	\SmallVBar
○	\Circle	▲	\FilledSmallTriangleUp	↓	\SpinDown
×	\Cross	■	\FilledSquare	↑	\SpinUp
◈	\DiamondShadowA	■	\FilledSquareShadowA	□	\Square
◈	\DiamondShadowB	■	\FilledSquareShadowC	▣	\SquareShadowA
◈	\DiamondShadowC	▼	\FilledTriangleDown	▣	\SquareShadowB
◇	\Diamondshape	◀	\FilledTriangleLeft	▣	\SquareShadowC
●	\FilledBigCircle	▶	\FilledTriangleRight	▽	\TriangleDown
◆	\FilledBigDiamondshape	▲	\FilledTriangleUp	◁	\TriangleLeft
■	\FilledBigSquare	—	\HBar	▷	\TriangleRight
▼	\FilledBigTriangleDown	⬙	\LowerDiamond	△	\TriangleUp
◀	\FilledBigTriangleLeft	⬗	\RightDiamond	|	\VBar

The ifsym documentation points out that one can use \rlap to combine some of the above into useful, new symbols. For example, \BigCircle and \FilledSmallCircle combine to give "◉". Likewise, \Square and \Cross combine to give "☒". See Section 7.2 for more information about constructing new symbols out of existing symbols.

Table 177: bbding Geometric Shapes

○	\CircleShadow	▌	\Rectangle	❑	\SquareShadowTopLeft
●	\CircleSolid	▮	\RectangleBold	❒	\SquareShadowTopRight
◆	\DiamondSolid	▏	\RectangleThin	■	\SquareSolid
⬭	\Ellipse	□	\Square	▼	\TriangleDown
⬭	\EllipseShadow	❑	\SquareCastShadowBottomRight	▲	\TriangleUp
⬬	\EllipseSolid	❑	\SquareCastShadowTopLeft		
◖	\HalfCircleLeft	❒	\SquareCastShadowTopRight		
◗	\HalfCircleRight	❑	\SquareShadowBottomRight		

Table 178: pifont Geometric Shapes

●	\ding{108}	❑	\ding{111}	❒	\ding{114}	◆	\ding{117}	▏	\ding{121}
○	\ding{109}	❒	\ding{112}	▲	\ding{115}	◗	\ding{119}	▮	\ding{122}
■	\ding{110}	❑	\ding{113}	▼	\ding{116}	|	\ding{120}		

Table 179: universa Geometric Shapes

● \baucircle ■ \bausquare ▲ \bautriangle

Table 180: manfnt Dangerous Bend Symbols

\dbend \lhdbend \reversedvideodbend

Note that these symbols descend far beneath the baseline. manfnt also defines non-descending versions, which it calls, correspondingly, \textdbend, \textlhdbend, and \textreversedvideodbend.

Table 181: skull Symbols

\skull

Table 182: Non-Mathematical mathabx Symbols

\rip

Table 183: marvosym Information Symbols

	\Bicycle		\Football		\Pointinghand
☑	\Checkedbox		\Gentsroom		\Wheelchair
	\Clocklogo		\Industry		\Writinghand
	\Coffeecup		\Info		
☒	\Crossedbox		\Ladiesroom		

Table 184: Miscellaneous dingbat Dingbats

⚓	\anchor	👁	\eye	▤	\Sborder
♪	\carriagereturn	❖	\filledsquarewithdots	⊡	\squarewithdots
✓	\checkmark	🛰	\satellitedish	▥	\Zborder

Table 185: Miscellaneous bbding Dingbats

✉	\Envelope	✌	\Peace	✆	\PhoneHandset	✪	\SunshineOpenCircled
❖	\OrnamentDiamondSolid	☎	\Phone	✈	\Plane	☮	\Tape

Table 186: Miscellaneous pifont Dingbats

☎	\ding{37}	✈	\ding{40}	❤	\ding{164}	❧	\ding{167}	♠	\ding{171}
✆	\ding{38}	✉	\ding{41}	❥	\ding{165}	♣	\ding{168}	♦	\ding{169}
✇	\ding{39}	❖	\ding{118}	❦	\ding{166}	♥	\ding{170}		

6 Other symbols

The following are all the symbols that didn't fit neatly or unambiguously into any of the previous sections. (Do weather symbols belong under "Science and technology"? Should dice be considered "mathematics"?) While some of the tables contain clearly related groups of symbols (e.g., musical notes), others represent motley assortments of whatever the font designer felt like drawing.

Table 187: textcomp Genealogical Symbols

⋆	\textborn	⚮	\textdivorced	⚭	\textmarried
†	\textdied		\textleaf		

Table 188: wasysym General Symbols

	\ataribox	⏲	\clock	◀	\LEFTarrow	☺	\smiley
	\bell	⌀	\diameter	ϟ	\lightning	☼	\sun
☻	\blacksmiley	▼	\DOWNarrow	☎	\phone	▲	\UParrow
⋈	\Bowtie	☹	\frownie	➪	\pointer	⌑	\wasylozenge
¦	\brokenvert	⦸	\invdiameter	⌕	\recorder		
✓	\checked	✠	\kreuz	▶	\RIGHTarrow		

Table 189: wasysym Musical Notes

♪	\eighthnote	𝅗𝅥	\halfnote	♫	\twonotes	𝅝	\fullnote	♩	\quarternote

See also \flat, \sharp, and \natural (Table 125 on page 38).

Table 190: wasysym Circles

●	\CIRCLE	◐	\LEFTcircle	◑	\RIGHTcircle	↻	\rightturn
○	\Circle	◖	\Leftcircle	◗	\Rightcircle		
◖	\LEFTCIRCLE	◗	\RIGHTCIRCLE	↺	\leftturn		

Table 191: Miscellaneous manfnt Symbols

	\manboldkidney		\manpenkidney
	\manconcentriccircles		\manquadrifolium
	\manconcentricdiamond		\manquartercircle
	\mancone		\manrotatedquadrifolium
	\mancube		\manrotatedquartercircle
	\manerrarrow		\manstar
	\manfilledquartercircle		\mantiltpennib
	\manhpennib	▼	\mantriangledown
	\manimpossiblecube	▶	\mantriangleright
	\mankidney	▲	\mantriangleup
	\manlhpenkidney		\manvpennib

Table 192: marvosym Navigation Symbols

▶	\Forward	▼	\MoveDown	⏮	\RewindToIndex	⤒	\ToTop
⏭	\ForwardToEnd	▲	\MoveUp	⏮	\RewindToStart		
⏭	\ForwardToIndex	◀	\Rewind	⤓	\ToBottom		

Table 193: marvosym Laundry Symbols

40	\AtForty		\Handwash	95	\ShortNinetyFive
95	\AtNinetyFive		\IroningI	60	\ShortSixty
60	\AtSixty		\IroningII	30	\ShortThirty
△	\Bleech		\IroningIII	40	\SpecialForty
Ⓐ	\CleaningA		\NoBleech		\Tumbler
Ⓕ	\CleaningF		\NoChemicalCleaning		\WashCotton
Ⓕ	\CleaningFF		\NoIroning		\WashSynthetics
Ⓟ	\CleaningP		\NoTumbler		\WashWool
Ⓟ	\CleaningPP	50	\ShortFifty		
	\Dontwash	40	\ShortForty		

Table 194: Other marvosym Symbols

☥	\Ankh	✝	\Cross	♡	\Heart	☺	\Smiley
	\Bat		\FHBOlogo		\MartinVogel		\Womanface
	\Bouquet		\FHBOLOGO		\Mundus		\Yinyang
✞	\Celtcross		\Frowny	@	\MVAt		
Ⓐ	\CircledA		\FullFHBO	➔	\Rightarrow*		

* Standard LaTeX 2_ε defines \Rightarrow to display "⇒", while marvosym redefines it to display "➔" (or ":" in math mode). This conflict can be problematic for math symbols defined in terms of \Rightarrow, such as \Longleftrightarrow, which ends up looking like "⇐:".

Table 195: Miscellaneous universa Symbols

\bauforms \bauhead

Table 196: ifsym Weather Symbols

\Cloud	\Hail	\Sleet	\WeakRain
\FilledCloud	\HalfSun	\Snow	\WeakRainCloud
\FilledRainCloud	\Lightning	\SnowCloud	\FilledSnowCloud
\FilledSunCloud	\NoSun	\Sun	
\FilledWeakRainCloud	\Rain	\SunCloud	
\Fog	\RainCloud	\ThinFog	

In addition, `\Thermo{0}`...`\Thermo{6}` produce thermometers that are between 0/6 and 6/6 full of mercury:

Similarly, `\wind{`⟨*sun*⟩`}{`⟨*angle*⟩`}{`⟨*strength*⟩`}` will draw wind symbols with a given amount of sun (0–4), a given angle (in degrees), and a given strength in km/h (0–100). For example, `\wind{0}{0}{0}` produces "", `\wind{2}{0}{0}` produces "", and `\wind{4}{0}{100}` produces "".

Table 197: ifsym Alpine Symbols

\SummitSign	\Summit	\SurveySign	\HalfFilledHut
\StoneMan	\Mountain	\Joch	\VarSummit
\Hut	\IceMountain	\Flag	
\FilledHut	\VarMountain	\VarFlag	
\Village	\VarIceMountain	\Tent	

Table 198: ifsym Clocks

\Interval	\StopWatchStart	\VarClock	\Wecker
\StopWatchEnd	\Taschenuhr	\VarTaschenuhr	

ifsym also exports a `\showclock` macro. `\showclock{`⟨*hours*⟩`}{`⟨*minutes*⟩`}` outputs a clock displaying the corresponding time. For instance, "`\showclock{5}{40}`" produces "". ⟨*hours*⟩ must be an integer from 0 to 11, and ⟨*minutes*⟩ must be an integer multiple of 5 from 0 to 55.

Table 199: Other ifsym Symbols

\FilledSectioningDiamond	\Letter	\Radiation
\Fire	\PaperLandscape	\SectioningDiamond
\Irritant	\PaperPortrait	\Telephone
\StrokeOne	\StrokeThree	\StrokeFive
\StrokeTwo	\StrokeFour	

In addition, `\Cube{1}`...`\Cube{6}` produce dice with the corresponding number of spots:

TABLE 200: skak Chess Informator Symbols

∓	\bbetter	⩲	\doublepawns	N	\novelty	R	\various
−+	\bdecisive	⊥	\ending	□	\onlymove	±	\wbetter
⌓	\betteris	=	\equal	◪	\opposbishops	+−	\wdecisive
◫	\bishoppair	‖	\etc	♗	\passedpawn	⨯	\weakpt
∓	\bupperhand	⇔	\file	≪	\qside	⌟	\with
⊞	\centre	≫	\kside	▞	\samebishops	→	\withattack
RR	\comment	✕	\markera	—	\see	△	\withidea
⩱	\compensation	○	\markerb	○⇢○	\seppawns	↑	\withinit
⇆	\counterplay	#	\mate	⊕	\timelimit	∟	\without
↻	\devadvantage	>	\morepawns	∞	\unclear	±	\wupperhand
⇗	\diagonal	◯	\moreroom	⩯	\unitedpawns	⊙	\zugzwang

The above symbols are merely the named informator symbol. skak can typeset many more chess-related symbols, including those for all of the pieces (♔♕♖♗♘♙/♚♛♜♝♞♟), but only in the context of moves and boards, not as individual, named LaTeX symbols.

7 Additional Information

Unlike the previous sections of this document, Section 7 does not contain new symbol tables. Rather, it provides additional help in using the Comprehensive LaTeX Symbol List. First, it draws attention to symbol names used by multiple packages. Next, it provides some guidelines for finding symbols and gives some examples regarding how to construct missing symbols out of existing ones. Then, it comments on the spacing surrounding symbols in math mode. After that, it presents an ASCII and Latin 1 quick-reference guide, showing how to enter all of the standard ASCII/Latin 1 symbols in LaTeX. And finally, it lists some statistics about this document itself.

7.1 Symbol Name Clashes

Unfortunately, a number of symbol names are not unique; they appear in more than one package. Depending on how the symbols are defined in each package, LaTeX will either output an error message or replace an earlier-defined symbol with a later-defined symbol. Table 201 presents a selection of name clashes that appear in this document.

Using multiple symbols with the same name in the same document—or even merely loading conflicting symbol packages—can be tricky, but, as evidenced by the existence of Table 201, not impossible. The general procedure is to load the first package, rename the conflicting symbols, and then load the second package. Examine the LaTeX source for this document (`symbols.tex`)—especially the `\savesymbol` and `\restoresymbol` macros and their subsequent usage—to see one possible way to handle symbol conflicts.

txfonts and pxfonts redefine a huge number of symbols—essentially, all of the symbols defined by latexsym, textcomp, the various $\mathcal{AMS}$ symbol sets, and LaTeX 2_ε itself. Similarly, mathabx redefines a vast number of math symbols in an attempt to improve their look. The txfonts, pxfonts, and mathabx conflicts are not listed in Table 201 because they are designed to be compatible with the symbols they replace. Table 202 on page 57 illustrates what "compatible" means in this context.

To use the new txfonts/pxfonts symbols without altering the document's main font, merely reset the default font families back to their original values after loading one of those packages:

```
\renewcommand\rmdefault{cmr}
\renewcommand\sfdefault{cmss}
\renewcommand\ttdefault{cmtt}
```

7.2 Where can I find the symbol for ... ?

If you can't find some symbol you're looking for in this document, there are a few possible explanations:

- The symbol isn't intuitively named. As a few examples, the command to draw dice is "`\Cube`"; a plus sign with a circle around it ("exclusive or" to computer engineers) is "`\oplus`"; and lightning bolts in fonts designed by German speakers may have "blitz" in their names. The moral of the story is to be creative with synonyms when searching the index.
- The symbol is defined by some package that I overlooked (or deemed unimportant). If there's some symbol package that you think should be included in the Comprehensive LaTeX Symbol List, please send me e-mail at the address listed on the title page.
- The symbol isn't defined in any package whatsoever.

Even in the last case, all is not lost. Sometimes, a symbol exists in a font, but there is no LaTeX binding for it. For example, the PostScript Symbol font contains a "↵" symbol, which may be useful for representing a carriage return, but there is no package for accessing that symbol (as far as I know). To produce an unnamed symbol, you need to switch to the font explicitly with LaTeX 2_ε's low-level font commands [LaT00] and use TeX's primitive `\char` command [Knu86a] to request a specific character number in the font.[3] In fact, `\char` is not strictly necesssary; the character can often be entered symbolically. For example, the symbol for a Tate-Shafarevich group ("Ш") is actually an uppercase *sha* in the Cyrillic alphabet. (Cyrillic is supported by the OT2 font encoding, for instance). While a *sha* can be defined numerically as "`{\fontencoding{OT2}\selectfont\char88}`" it may be more intuitive to use the OT2 font encoding's "SH" ligature: "`{\fontencoding{OT2}\selectfont SH}`".

[3] pifont defines a convenient `\Pisymbol` command for accessing symbols in PostScript fonts by number. For example, "`\Pisymbol{psy}{191}`" produces "↵".

Table 201: Symbol Name Clashes

Symbol	$\LaTeX\,2_\varepsilon$	$\mathcal{AMS}$	stmaryrd	wasysym	mathabx	marvosym	bbding	ifsym	dingbat	wsuipa
\baro			⦶							ɵ
\bigtriangledown	▽		▽							
\bigtriangleup	△		△							
\checkmark		✓							✔	
\Circle				○				○		
\Cross						✝	✝	×		
\ggg		⋙			⋙					
\Letter						✉		✉		
\lightning			↯	↯						
\Lightning						↯		☇		
\lll		⋘			⋘					
\Rightarrow	⇒				⇒	➔				
\Square				□			□	□		
\Sun					⊙	☉		☼		
\TriangleDown							▼	▽		
\TriangleUp							▲	△		

TABLE 202: Example of a Benign Name Clash

Symbol	Default (Computer Modern)	txfonts (Times Roman)
`R`	R	R
`\textrecipe`	℞	℞

Reflecting and rotating existing symbols

A common request on `comp.text.tex` is for a reversed or rotated version of an existing symbol. As a last resort, these effects can be achieved with the graphicx (or graphics) package's `\reflectbox` and `\rotatebox` macros. For example, `\rotatebox[origin=c]{180}{$\iota$}` produces the definite-description operator ("$\reflectbox{\rotatebox{180}{\iota}}$"). The disadvantage of the graphicx/graphics approach is that not every TEX backend handles graphical transformations.[4] Far better is to find a suitable font that contains the desired symbol in the correct orientation. For instance, if the phonetic package is available, then `\textit{\riota}` will yield a backend-independent "$\textit{\riota}$". Similarly, tipa's `\textrevepsilon` ("ɜ") or wsuipa's `\revepsilon` ("ɜ") may be used to express the mathematical notion of "such that" in a cleaner manner than with `\reflectbox` or `\rotatebox`.

Joining and overlapping existing symbols

Symbols that do not exist in any font can sometimes be fabricated out of existing symbols. The LaTeX 2_ε source file `fontdef.dtx` contains a number of such definitions. For example, `\models` (see Table 42 on page 21) is defined in that file with:

```
\def\models{\mathrel|\joinrel=}
```

where `\mathrel` and `\joinrel` are used to control the horizontal spacing. `\def` is the TEX primitive upon which LaTeX's `\newcommand` is based. See The TEXbook [Knu86a] for more information on all three of those commands.

With some simple pattern-matching, one can easily define a backward `\models` sign ("=|"):

```
\def\ismodeledby{=\joinrel\mathrel|}
```

In general, arrows/harpoons, horizontal lines ("=", "-", "`\relbar`", and "`\Relbar`"), and the various math-extension characters can be combined creatively with miscellaneous other characters to produce a variety of new symbols. Of course, new symbols can be composed from *any* set of existing characters. For instance, LaTeX defines `\hbar` ("$\hbar$") as a "$^-$" character (`\mathchar'26`) followed by a backspace of 9 math units (`\mkern-9mu`), followed by the letter "h":

```
\def\hbar{{\mathchar'26\mkern-9muh}}
```

We can just as easily define other barred letters:

```
\def\bbar{{\mathchar'26\mkern-9mu b}}
\def\dbar{{\mathchar'26\mkern-12mu d}}
```

(The space after the "mu" is optional but is added for clarity.) `\bbar` and `\dbar` define "$\bbar$" and "$\dbar$", respectively. Note that `\dbar` requires a greater backward math kern than `\bbar`; a −9 mu kern would have produced the less-attractive "$\overline{d}$" glyph.

There is a TEX primitive called `\mathaccent` which centers one mathematical symbol atop another. For example, one can define `\dotcup` ("$\dotcup$")—the composition of a `\cup` and a `\cdot`—as follows:

```
\newcommand{\dotcup}{\ensuremath{\mathaccent\cdot\cup}}
```

[4] As an example, Xdvi ignores both `\reflectbox` and `\rotatebox`.

The catch is that `\mathaccent` requires the accent to be a "math character". That is, it must be a character in a math font as opposed to a symbol defined in terms of other symbols. See The TEXbook [Knu86a] for more information.

The slashed package, although originally designed for producing Feynman slashed-character notation, in fact facilitates the production of *arbitrary* overlapped symbols. The default behavior is to overwrite a given character with "/". For example, `\slashed{D}` produces "$\not{D}$". However, the `\declareslashed` command provides the flexibility to specify the mathematical context of the composite character (operator, relation, punctuation, etc., as will be discussed in Section 7.3), the overlapping symbol, horizontal and vertical adjustments in symbol-relative units, and the character to be overlapped. Consider, for example, the symbol for reduced quadrupole moment ("$\not{I}$"). This can be declared as follows:

```
\newcommand{\rqm}{{%
  \declareslashed{}{\text{-}}{0.04}{0}{I}\slashed{I}}}
```

`\declareslashed{·}{·}{·}{·}{I}` affects the meaning of all subsequent `\slashed{I}` commands in the same scope. The preceding definition of `\rqm` therefore uses an extra set of curly braces to limit that scope to a single `\slashed{I}`. In addition, `\rqm` uses amstext's `\text` macro (described on the next page) to make `\declareslashed` use a text-mode hyphen ("-") instead of a math-mode minus sign ("−") and to ensure that the hyphen scales properly in size in subscripts and superscripts. See slashed's documentation (located in `slashed.sty` itself) for a detailed usage description of the `\slashed` and `\declareslashed` commands.

Making new symbols work in superscripts and subscripts

To make composite symbols work properly within subscripts and superscripts, you may need to use TEX's `\mathchoice` primitive. `\mathchoice` evaluates one of four expressions, based on whether the current math style is display, text, script, or scriptscript. (See The TEXbook [Knu86a] for a more complete description.) For example, the following LATEX code—posted to `comp.text.tex` by Torsten Bronger—composes a sub/superscriptable "⊥" symbol out of `\top` and `\bot` ("⊤" and "⊥"):

```
\def\topbotatom#1{\hbox{\hbox to 0pt{$#1\bot$\hss}$#1\top$}}
\newcommand*{\topbot}{\mathrel{\mathchoice{\topbotatom\displaystyle}
                                  {\topbotatom\textstyle}
                                  {\topbotatom\scriptstyle}
                                  {\topbotatom\scriptscriptstyle}}}
```

The following is another example that uses `\mathchoice` to construct symbols in different math modes. The code defines a principal value integral symbol, which is an integral sign with a line through it.

```
\def\Xint#1{\mathchoice
   {\XXint\displaystyle\textstyle{#1}}%
   {\XXint\textstyle\scriptstyle{#1}}%
   {\XXint\scriptstyle\scriptscriptstyle{#1}}%
   {\XXint\scriptscriptstyle\scriptscriptstyle{#1}}%
   \!\int}
\def\XXint#1#2#3{{\setbox0=\hbox{$#1{#2#3}{\int}$}
     \vcenter{\hbox{$#2#3$}}\kern-.5\wd0}}
\def\ddashint{\Xint=}
\def\dashint{\Xint-}
```

(The preceding code was taken verbatim from the UK TEX Users' Group FAQ at `http://www.tex.ac.uk/faq`.) `\dashint` produces a single-dashed integral sign ("⨍"), while `\ddashint` produces a double-dashed one ("⨎"). The `\Xint` macro defined above can also be used to generate a wealth of new integrals: "∲" (`\Xint\circlearrowright`), "∳" (`\Xint\circlearrowleft`), "⨏" (`\Xint\subset`), "⨜" (`\Xint\infty`), and so forth.

LATEX 2ε provides a simple wrapper for `\mathchoice` that sometimes helps produce terser symbol definitions. The macro is called `\mathpalette` and it takes two arguments. `\mathpalette` invokes the first argument, passing it one of "`\displaystyle`", "`\textstyle`", "`\scriptstyle`", or "`\scriptscriptstyle`", followed by the second argument. `\mathpalette` is useful when a symbol macro must know which math style is currently in use (e.g., to set it explicitly within an `\mbox`). Donald Arseneau posted the following `\mathpalette`-based definition of a probabilistic-independence symbol ("⫫") to `comp.text.tex`:

```
\newcommand\independent{\protect\mathpalette{\protect\independenT}{\perp}}
\def\independenT#1#2{\mathrel{\rlap{$#1#2$}\mkern2mu{#1#2}}}
```

The `\independent` macro uses `\mathpalette` to pass the `\independenT` helper macro both the current math style and the `\perp` symbol. `\independenT` typesets `\perp` in the current math style, moves two math units to the right, and finally typesets a second—overlapping—copy of `\perp`, again in the current math style. `\rlap`, which enables text overlap, is described later on this page.

Some people like their square-root signs with a trailing "hook" (i.e., "$\sqrt{\ }$") as this helps visually distinguish expressions like "$\sqrt{3x}$" from those like "$\sqrt{3}x$". Dan Luecking posted a `\mathpalette`-based definition of a hooked square-root symbol to `comp.text.tex`:

```
\def\hksqrt{\mathpalette\DHLhksqrt}
\def\DHLhksqrt#1#2{\setbox0=\hbox{$#1\sqrt{#2\,}$}\dimen0=\ht0
  \advance\dimen0-0.2\ht0
  \setbox2=\hbox{\vrule height\ht0 depth -\dimen0}%
  {\box0\lower0.4pt\box2}}
```

Notice how `\DHLhksqrt` uses `\mathpalette` to recover the outer math style (argument `#1`) from within an `\hbox`. The rest of the code is simply using TEX primitives to position a hook of height 0.2 times the `\sqrt` height at the right of the `\sqrt`. See The TEXbook [Knu86a] for more understanding of TEX "boxes" and "dimens".

Sometimes, however, amstext's `\text` macro is all that is necessary to make composite symbols appear correctly in subscripts and superscripts, as in the following definitions of `\neswarrow` ("$\swarrow$") and `\nwsearrow` ("$\nwarrow$"):[5]

```
\newcommand{\neswarrow}{\mathrel{\text{$\nearrow$\llap{$\swarrow$}}}}
\newcommand{\nwsearrow}{\mathrel{\text{$\nwarrow$\llap{$\searrow$}}}}
```

`\text` resembles LATEX's `\mbox` command but shrinks its argument appropriately when used within a subscript or superscript. `\llap` ("left overlap") and its counterpart, `\rlap` ("right overlap"), appear frequently when creating composite characters. `\llap` outputs its argument to the left of the current position, overlapping whatever text is already there. Similarly, `\rlap` overlaps whatever text would normally appear to the right of its argument. For example, "`A\llap{B}`" and "`\rlap{A}B`" each produce "B". However, the result of the former is the width of "A", and the result of the latter is the width of "B"—`\llap{...}` and `\rlap{...}` take up zero space.

In a June 2002 post to `comp.text.tex`, Donald Arseneau presented a general macro for aligning an arbitrary number of symbols on their horizontal centers and vertical baselines:

```
\makeatletter
  \def\moverlay{\mathpalette\mov@rlay}
  \def\mov@rlay#1#2{\leavevmode\vtop{%
     \baselineskip\z@skip \lineskiplimit-\maxdimen
     \ialign{\hfil$#1##$\hfil\cr#2\crcr}}}
\makeatother
```

`\moverlay` takes a list of symbols separated by `\cr` (TEX's equivalent of LATEX's `\\`). For example, the `\topbot` command defined on the previous page could have been expressed as "`\moverlay{\top\cr\bot}`" and the `\neswarrow` command defined above could have been expressed as "`\moverlay{\nearrow\cr\swarrow}`".

The basic concept behind `\moverlay`'s implementation is that `\moverlay` typesets the given symbols in a table that utilizes a zero `\baselineskip`. This causes every row to be typeset at the same vertical position. See The TEXbook [Knu86a] for explanations of the TEX primitives used by `\moverlay`.

Modifying LATEX-generated symbols

Oftentimes, symbols composed in the LATEX 2_ε source code can be modified with minimal effort to produce useful variations. For example, `fontdef.dtx` composes the `\ddots` symbol (see Table 121 on page 37) out of three periods, raised 7 pt., 4 pt., and 1 pt., respectively:

[5] Note that if your goal is to typeset commutative diagrams, then you should probably be using XY-pic.

```
\def\ddots{\mathinner{\mkern1mu\raise7\p@
    \vbox{\kern7\p@\hbox{.}}\mkern2mu
    \raise4\p@\hbox{.}\mkern2mu\raise\p@\hbox{.}\mkern1mu}}
```

`\p@` is a LaTeX 2_ε shortcut for "`pt`" or "`1.0pt`". The remaining commands are defined in The TeXbook [Knu86a]. To draw a version of `\ddots` with the dots going along the opposite diagonal, we merely have to reorder the `\raise7\p@`, `\raise4\p@`, and `\raise\p@`:

```
\makeatletter
  \def\revddots{\mathinner{\mkern1mu\raise\p@
    \vbox{\kern7\p@\hbox{.}}\mkern2mu
    \raise4\p@\hbox{.}\mkern2mu\raise7\p@\hbox{.}\mkern1mu}}
\makeatother
```

The `\makeatletter` and `\makeatother` commands are needed to coerce LaTeX into accepting "`@`" as part of a macro name. `\revddots` is essentially identical to the mathdots package's `\iddots` command or the yhmath package's `\adots` command.

Producing complex accents

Accents are a special case of combining existing symbols to make new symbols. While various tables in this document show how to add an accent to an existing symbol, some applications, such as transliterations from non-Latin alphabets, require *multiple* accents per character. For instance, the creator of pdfTeX writes his name as "Hàn Thế́ Thành". The wsuipa package defines `\diatop` and `\diaunder` macros for putting one or more diacritics or accents above or below a given character. For example, `\diaunder[{\diatop[\'|\=]}|\textsubdot{r}]` produces "$\acute{\underset{\cdot}{\text{r}}}$". See the wsuipa documentation for more information.

The accents package facilitates the fabrication of accents in math mode. Its `\accentset` command enables *any* character to be used as an accent. For instance, `\accentset{\star}{f}` produces "$\overset{\star}{f}$" and `\accentset{e}{X}` produces "$\overset{e}{X}$". `\underaccent` does the same thing, but places the accent beneath the character. This enables constructs like `\underaccent{\tilde}{V}`, which produces "$\underset{\sim}{V}$". accents provides other accent-related features as well; see the documentation for more information.

A more complex example of composing accents is the following definition of extensible `\overbracket`, `\underbracket`, `\overparenthesis`, and `\underparenthesis` symbols, taken from a `comp.text.tex` post by Donald Arseneau:

```
\makeatletter
\def\overbracket#1{\mathop{\vbox{\ialign{##\crcr\noalign{\kern3\p@}
      \downbracketfill\crcr\noalign{\kern3\p@\nointerlineskip}
      $\hfil\displaystyle{#1}\hfil$\crcr}}}\limits}
\def\underbracket#1{\mathop{\vtop{\ialign{##\crcr
      $\hfil\displaystyle{#1}\hfil$\crcr\noalign{\kern3\p@\nointerlineskip}
      \upbracketfill\crcr\noalign{\kern3\p@}}}}\limits}
\def\overparenthesis#1{\mathop{\vbox{\ialign{##\crcr\noalign{\kern3\p@}
      \downparenthfill\crcr\noalign{\kern3\p@\nointerlineskip}
      $\hfil\displaystyle{#1}\hfil$\crcr}}}\limits}
\def\underparenthesis#1{\mathop{\vtop{\ialign{##\crcr
      $\hfil\displaystyle{#1}\hfil$\crcr\noalign{\kern3\p@\nointerlineskip}
      \upparenthfill\crcr\noalign{\kern3\p@}}}}\limits}
\def\downparenthfill{$\m@th\braceld\leaders\vrule\hfill\bracerd$}
\def\upparenthfill{$\m@th\bracelu\leaders\vrule\hfill\braceru$}
\def\upbracketfill{$\m@th\makesm@sh{\llap{\vrule\@height3\p@\@width.7\p@}}%
  \leaders\vrule\@height.7\p@\hfill
  \makesm@sh{\rlap{\vrule\@height3\p@\@width.7\p@}}$}
\def\downbracketfill{$\m@th
  \makesm@sh{\llap{\vrule\@height.7\p@\@depth2.3\p@\@width.7\p@}}%
  \leaders\vrule\@height.7\p@\hfill
  \makesm@sh{\rlap{\vrule\@height.7\p@\@depth2.3\p@\@width.7\p@}}$}
\makeatother
```

Table 203 showcases these accents. The TEXbook [Knu86a] or another book on TEX primitives is indispensible for understanding how the preceding code works. The basic idea is that `\downparenthfill`, `\upparenthfill`, `\downbracketfill`, and `\upbracketfill` do all of the work; they output a left symbol (e.g., `\braceld` [“⎩”] for `\downparenthfill`), a horizontal rule that stretches as wide as possible, and a right symbol (e.g., `\bracerd` [“⎭”] for `\downparenthfill`). `\overbracket`, `\underbracket`, `\overparenthesis`, and `\underparenthesis` merely create a table whose width is determined by the given text, thereby constraining the width of the horizontal rules.

TABLE 203: Manually Composed Extensible Accents

$\overbracket{abc}$	`\overbracket{abc}`	$\overparenthesis{abc}$	`\overparenthesis{abc}`
$\underbracket{abc}$	`\underbracket{abc}`	$\underparenthesis{abc}$	`\underparenthesis{abc}`

A similar, but simpler example, stems from another `comp.text.tex` post by Donald Arseneau. The following code defines an equals sign that extends as far to the right as possible (just like LATEX's `\hrulefill` command):

```
\makeatletter
\def\equalsfill{$\m@th\mathord=\mkern-7mu
  \cleaders\hbox{$\!\mathord=\!$}\hfill
  \mkern-7mu\mathord=$}
\makeatother
```

TEX's `\cleaders` and `\hfill` primitives are the key to understanding `\equalsfill`'s extensibility. Essentially, `\equalsfill` repeats a box containing “=” plus some negative space until it fills the maximum available horizontal space. `\equalsfill` is intended to be used with LATEX's `\stackrel` command, which stacks one mathematical expression (slightly reduced in size) atop another. Hence, “`\stackrel{a}{\rightarrow}`” produces “$\stackrel{a}{\rightarrow}$” and “X `\stackrel{\text{definition}}{\hbox{\equalsfill}}` Y” produces “$X \stackrel{\text{definition}}{=======} Y$”.

If all that needs to extend are horizontal and vertical lines—as opposed to repeated symbols such as the “=” in the previous example—LATEX's `array` or `tabular` environments may suffice. Consider the following code (also presented in a `comp.text.tex` post by Donald Arseneau) for typesetting annuities:

```
\DeclareRobustCommand{\annu}[1]{_{%
 \def\arraystretch{0}%
 \setlength\arraycolsep{1pt}%        adjust these
 \setlength\arrayrulewidth{.2pt}%  two settings
 \begin{array}[b]{@{}c|}\hline
 \\[\arraycolsep]%
 \scriptstyle #1%
 \end{array}%
}}
```

One can then use, e.g., “`$A\annu{x:n}$`” to produce “$A_{\overline{x:n}|}$”.

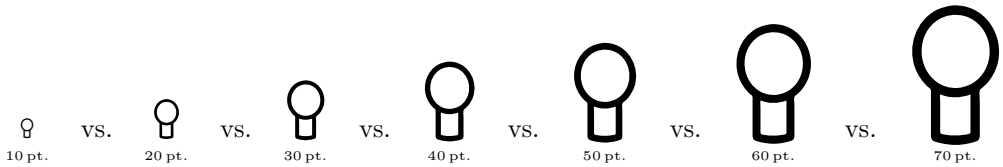
Creating new symbols from scratch

Sometimes is it simply not possible to define a new symbol in terms of existing symbols. Fortunately, most, if not all, TEX distributions are shipped with a tool called METAFONT which is designed specifically for creating fonts to be used with TEX. The METAFONTbook [Knu86b] is the authoritative text on METAFONT. If you plan to design your own symbols with METAFONT, The METAFONTbook is essential reading. Nevertheless, the following is an extremely brief tutorial on how to create a new LATEX symbol using METAFONT. Its primary purpose is to cover the LATEX-specific operations not mentioned in The METAFONTbook and to demonstrate that symbol-font creation is not necessarily a difficult task.

Suppose we need a symbol to represent a light bulb (“💡”).[6] The first step is to draw this in METAFONT. It is common to separate the font into two files: a size-dependent file, which specifies the design size and

[6] I'm not a very good artist; you'll have to pretend that “💡” looks like a light bulb.

various font-specific parameters that are a function of the design size; and a size-independent file, which draws characters in the given size. Figure 1 shows the METAFONT code for `lightbulb10.mf`. `lightbulb10.mf` specifies various parameters that produce a 10 pt. light bulb then loads `lightbulb.mf`. Ideally, one should produce `lightbulb`⟨*size*⟩`.mf` files for a variety of ⟨*size*⟩s. This is called "optical scaling". It enables, for example, the lines that make up the light bulb to retain the same thickness at different font sizes, which looks much nicer than the alternative—and default—"mechanical scaling". When a `lightbulb`⟨*size*⟩`.mf` file does not exist for a given size ⟨*size*⟩, the computer mechanically produces a wider, taller, thicker symbol:




```
font_identifier := "LightBulb10";                    % Name the font.
font_size 10pt#;                                     % Specify the design size.

em# := 10pt#;                                        % "M" width is 10 points.
cap# := 7pt#;                                        % Capital letter height is 7 points above the baseline.
sb# := 1/4pt#;                                       % Leave this much space on the side of each character.
o# := 1/16pt#;                                       % Amount that curves overshoot borders.

input lightbulb                                      % Load the file that draws the actual glyph.
```

Figure 1: Sample METAFONT size-specific file (`lightbulb10.mf`)

`lightbulb.mf`, shown in Figure 2, draws a light bulb using the parameters defined in `lightbulb10.mf`. Note that the the filenames "`lightbulb10.mf`" and "`lightbulb.mf`" do not follow the Berry font-naming scheme [Ber01]; the Berry font-naming scheme is largely irrelevant for symbol fonts, which generally lack bold, italic, small-caps, slanted, and other such variants.

The code in Figures 1 and 2 is heavily commented and should demonstrate some of the basic concepts behind METAFONT usage: declaring variables, defining points, drawing lines and curves, and preparing to debug or fine-tune the output. Again, The METAFONTbook [Knu86b] is the definitive reference on METAFONT programming.

METAFONT can produce "proofs" of fonts—large, labeled versions that showcase the logical structure of each character. In fact, proof mode is METAFONT's default mode. To produce a proof of `lightbulb10.mf`, issue the following commands at the operating-system prompt:

```
prompt> mf lightbulb10.mf                    ⇐ Produces lightbulb10.2602gf
prompt> gftodvi lightbulb10.2602gf           ⇐ Produces lightbulb10.dvi
```

You can then view `lightbulb10.dvi` with any DVI viewer. The result is shown in Figure 3. Observe how the grid defined with *makegrid* at the bottom of Figure 2 draws vertical lines at positions 0, *sb*, $w/2$, and $w - sb$ and horizontal lines at positions 0, $-1pt$, y_2, and h. Similarly, observe how the *penlabels* command labels all of the important coordinates: $z_1, z_2, \ldots, z_8$ and z_{67}, which `lightbulb.mf` defines to lie between z_6 and z_7.

Most, if not all, TEX distributions include a Plain TEX file called `testfont.tex` which is useful for testing new fonts in a variety of ways. One useful routine produces a table of all of the characters in the font:

```
prompt> tex testfont
This is TeX, Version 3.14159 (Web2C 7.3.1)
(/usr/share/texmf/tex/plain/base/testfont.tex
Name of the font to test = lightbulb10
Now type a test command (\help for help):)
*\table

*\bye
[1]
Output written on testfont.dvi (1 page, 1516 bytes).
```

```
mode_setup;                                          % Target a given printer.

define_pixels(em, cap, sb);                          % Convert to device-specific units.
define_corrected_pixels(o);                          % Same, but add a device-specific fudge factor.

%% Define a light bulb at the character position for "A"
%% with width 1/2em#, height cap#, and depth 1pt#.
beginchar("A", 1/2em#, cap#, 1pt#); "A light bulb";
    pickup pencircle scaled 1/2pt;                   % Use a pen with a small, circular tip.

    %% Define the points we need.
    top z1 = (w/2, h + o);                           % z1 is at the top of a circle.
    rt z2 = (w + sb + o - x4, y4);                   % z2 is at the same height as z4 but the opposite side.
    bot z3 = (z1 - (0, w - sb - o));                 % z3 is at the bottom of the circle.
    lft z4 = (sb - o, 1/2[y1, y3]);                  % z4 is on the left of the circle.
    path bulb;                                       % Define a path for the bulb itself.
    bulb = z1 .. z2 .. z3 .. z4 .. cycle;            % The bulb is a closed path.

    z5 = point 2 - 1/3 of bulb;                      % z5 lies on the bulb, a little to the right of z3.
    z6 = (x5, 0);                                    % z6 is at the bottom, directly under z5.
    z7 = (x8, 0);                                    % z7 is at the bottom, directly under z8.
    z8 = point 2 + 1/3 of bulb;                      % z8 lies on the bulb, a little to the left of z3.
    bot z67 = (1/2[x6, x7], pen_bot - o - 1/8pt); % z67 lies halfway between z6 and z7 but a jot lower.

    %% Draw the bulb and the base.
    draw bulb;                                       % Draw the bulb proper.
    draw z5 -- z6 .. z67 .. z7 -- z8;                % Draw the base of the bulb.

    %% Display key positions and points to help us debug.
    makegrid(0, sb, w/2, w - sb)(0, -1pt, y2, h);    % Label "interesting" x and y coordinates.
    penlabels(1, 2, 3, 4, 5, 6, 67, 7, 8);           % Label control points for debugging.
endchar;
end
```

Figure 2: Sample METAFONT size-independent file (`lightbulb.mf`)

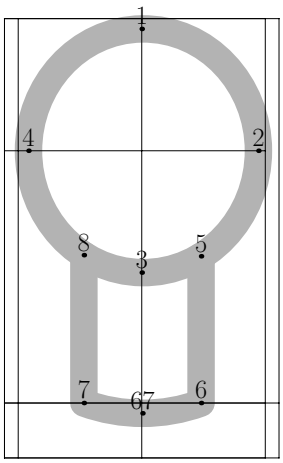


Figure 3: Proof diagram of `lightbulb10.mf`

```
Transcript written on testfont.log.
```

The resulting table, stored in `testfont.dvi` and illustrated in Figure 4, shows every character in the font. To understand how to read the table, note that the character code for "A"—the only character defined by `lightbulb10.mf`—is 41 in hexadecimal (base 16) and 101 in octal (base 8).

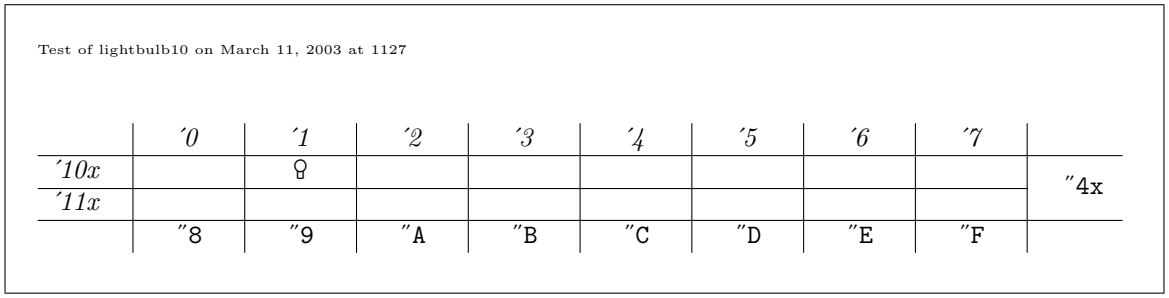


Figure 4: Font table produced by `testfont.tex`

The LightBulb10 font is now usable by TEX. LATEX 2_ε, however, needs more information before documents can use the font. First, we create a font-description file that tells LATEX 2_ε how to map fonts in a given font family and encoding to a particular font in a particular font size. For symbol fonts, this mapping is fairly simple. Symbol fonts almost always use the "U" ("Unknown") font encoding and frequently occur in only one variant: normal weight and non-italicized. The filename for a font-description file important; it must be of the form "⟨*encoding*⟩⟨*family*⟩`.fd`", where ⟨*encoding*⟩ is the lowercase version of the encoding name (typically "u" for symbol fonts) and ⟨*family*⟩ is the name of the font family. For LightBulb10, let's call this "bulb". Figure 5 lists the contents of `ubulb.fd`. The document "LATEX 2_ε Font Selection" [LAT00] describes `\DeclareFontFamily` and `\DeclareFontShape` in detail, but the gist of `ubulb.fd` is first to declare a `U`-encoded version of the `bulb` font family and then to specify that a LATEX 2_ε request for a `U`-encoded version of `bulb` with a (`m`)edium font series (as opposed to, e.g., bold) and a (`n`)ormal font shape (as opposed to, e.g., italic) should translate into a TEX request for `lightbulb10.tfm` mechanically scaled to the current font size.

```
\DeclareFontFamily{U}{bulb}{}
\DeclareFontShape{U}{bulb}{m}{n}{<-> lightbulb10}{}
```

Figure 5: LATEX 2_ε font-description file (`ubulb.fd`)

The final step is to write a LATEX 2_ε style file that defines a name for each symbol in the font. Because we have only one symbol our style file, `lightbulb.sty` (Figure 6), is rather trivial. Note that instead of typesetting "`A`" we could have had `\lightbulb` typeset "`\char65`", "`\char"41`", or "`\char'101`" (respectively, decimal, hexadecimal, and octal character offsets into the font). For a simple, one-character symbol font such as LightBulb10 it would be reasonable to merge `ubulb.fd` into `lightbulb.sty` instead of maintaining two separate files. In either case, a document need only include "`\usepackage{lightbulb}`" to make the `\lightbulb` symbol available.

```
\newcommand{\lightbulb}{{\usefont{U}{bulb}{m}{n}A}}
```

Figure 6: LATEX 2_ε style file (`lightbulb.sty`)

METAFONT normally produces bitmapped fonts. However, it is also possible, with the help of some external tools, to produce PostScript Type 1 fonts. These have the advantages of rendering better in Adobe® Acrobat® (at least in versions prior to 6.0) and of being more memory-efficient when handled by a PostScript interpreter. See `http://www.tex.ac.uk/cgi-bin/texfaq2html?label=textrace` for pointers to tools that can produce Type 1 fonts from METAFONT.

7.3 Math-mode spacing

Terms such as "binary operators", "relations", and "punctuation" in Section 3 primarily regard the surrounding spacing. (See the Short Math Guide for LATEX [Dow00] for a nice exposition on the subject.) To use a symbol

for a different purpose, you can use the TEX commands `\mathord`, `\mathop`, `\mathbin`, `\mathrel`, `\mathopen`, `\mathclose`, and `\mathpunct`. For example, if you want to use `\downarrow` as a variable (an "ordinary" symbol) instead of a delimiter, you can write "`$3 x + \mathord{\downarrow}$`" to get the properly spaced "$3x + \downarrow$" rather than the awkward-looking "$3x+ \downarrow$". Similarly, to create a dotted-union symbol ("$\dot\cup$") that spaces like the ordinary set-union symbol (`\cup`) it must be defined with `\mathbin`, just as `\cup` is. Contrast "`$A \dot{\cup} B$`" ("$A\dot\cup B$") with "`$A \mathbin{\dot{\cup}} B$`" ("$A\dot\cup B$"). See The TEXbook [Knu86a] for the definitive description of math-mode spacing.

The purpose of the "log-like symbols" in Tables 84 and 85 is to provide the correct amount of spacing around and within multiletter function names. Table 204 contrasts the output of the log-like symbols with various, naïve alternatives. In addition to spacing, the log-like symbols also handle subscripts properly. For example, "`\max_{p \in P}`" produces "$\max_{p\in P}$" in text, but "$\max\limits_{p\in P}$" as part of a displayed formula.

TABLE 204: Spacing Around/Within Log-like Symbols

LATEX expression	Output	
`$r \sin \theta$`	$r \sin \theta$	(best)
`$r sin \theta$`	$rsin\theta$	
`$r \mbox{sin} \theta$`	$r\mathrm{sin}\theta$	

The amsmath package makes it straightforward to define new log-like symbols:

```
\DeclareMathOperator{\atan}{atan}
\DeclareMathOperator*{\lcm}{lcm}
```

The difference between `\DeclareMathOperator` and `\DeclareMathOperator*` involves the handling of subscripts. With `\DeclareMathOperator*`, subscripts are written beneath log-like symbols in display style and to the right in text style. This is useful for limit operators (e.g., `\lim`) and functions that tend to map over a set (e.g., `\min`). In contrast, `\DeclareMathOperator` tells TEX that subscripts should always be displayed to the right of the operator, as is common for functions that take a single parameter (e.g., `\log` and `\cos`). Table 205 contrasts symbols declared with `\DeclareMathOperator` and `\DeclareMathOperator*` in both text style (`$...$`) and display style (`\[...\]`).[7]

TABLE 205: Defining new log-like symbols

Declaration function	`$\newlogsym_{p \in P}$`	`\[ \newlogsym_{p \in P} \]`
`\DeclareMathOperator`	$\mathrm{newlogsym}_{p\in P}$	$\mathrm{newlogsym}_{p\in P}$
`\DeclareMathOperator*`	$\mathrm{newlogsym}_{p\in P}$	$\mathrm{newlogsym}\limits_{p\in P}$

It is common to use a thin space (`\,`) between the words of a multiword operators, as in "`\DeclareMathOperator*{\argmax}{arg\,max}`". `\liminf`, `\limsup`, and all of the log-like symbols shown in Table 85 utilize this spacing convention.

7.4 Bold mathematical symbols

LATEX does not normally use bold symbols when typeseting mathematics. However, bold symbols are occasionally needed, for example when naming vectors. Any of the approaches described at `http://www.tex.ac.uk/cgi-bin/texfaq2html?label=boldgreek` can be used to produce bold mathematical symbols. Table 206 contrasts the output produced by these various techniques. As the table illustrates, these techniques exhibit variation in their formatting of Latin letters (upright vs. italic), formatting of Greek letters (bold vs. normal), formatting of operators and relations (bold vs. normal), and spacing.

[7] Note that `\displaystyle` can be used to force display style within `$...$` and `\textstyle` can be used to force text style within `\[...\]`.

TABLE 206: Producing bold mathematical symbols

Package	Code	Output	
none	`$\alpha + b = \Gamma \div D$`	$\alpha + b = \Gamma \div D$	(no bold)
none	`$\mathbf{\alpha + b = \Gamma \div D}$`	$\alpha + \mathbf{b} = \mathbf{\Gamma \div D}$	
none	`\boldmath$\alpha + b = \Gamma \div D$`	$\boldsymbol{\alpha + b = \Gamma \div D}$	
amsbsy	`$\pmb{\alpha + b = \Gamma \div D}$`	$\boldsymbol{\alpha + b = \Gamma \div D}$	(faked bold)
amsbsy	`$\boldsymbol{\alpha + b = \Gamma \div D}$`	$\boldsymbol{\alpha + b = \Gamma \div D}$	
bm	`$\bm{\alpha + b = \Gamma \div D}$`	$\boldsymbol{\alpha + b = \Gamma \div D}$	
fixmath	`$\mathbold{\alpha + b = \Gamma \div D}$`	$\boldsymbol{\alpha + b = \Gamma \div D}$	

7.5 ASCII and Latin 1 quick reference

Table 207 amalgamates data from various other tables in this document into a convenient reference for LaTeX 2ε typesetting of ASCII characters, i.e., the characters available on a typical U.S. computer keyboard. The first two columns list the character's ASCII code in decimal and hexadecimal. The third column shows what the character looks like. The fourth column lists the LaTeX 2ε command to typeset the character as a text character. And the fourth column lists the LaTeX 2ε command to typeset the character within a `\texttt{...}` command (or, more generally, when `\ttfamily` is in effect).

TABLE 207: LaTeX 2ε ASCII Table

Dec	Hex	Char	Body text	\texttt
33	21	!	!	!
34	22	"	\textquotedbl	"
35	23	#	\#	\#
36	24	$	\$	\$
37	25	%	\%	\%
38	26	&	\&	\&
39	27	’	’	’
40	28	(	(	(
41	29	)	)	)
42	2A	*	*	*
43	2B	+	+	+
44	2C	,	,	,
45	2D	-	-	-
46	2E	.	.	.
47	2F	/	/	/
48	30	0	0	0
49	31	1	1	1
50	32	2	2	2
⋮	⋮	⋮	⋮	⋮
57	39	9	9	9
58	3A	:	:	:
59	3B	;	;	;
60	3C	<	\textless	<
61	3D	=	=	=
62	3E	>	\textgreater	>
63	3F	?	?	?
64	40	@	@	@
65	41	A	A	A
66	42	B	B	B
67	43	C	C	C
⋮	⋮	⋮	⋮	⋮
90	5A	Z	Z	Z
91	5B	[	[	[
92	5C	\	\textbackslash	\char‘\\
93	5D	]	]	]
94	5E	^	\^{}	\^{}
95	5F	_	_	\char‘_
96	60	‘	‘	‘
97	61	a	a	a
98	62	b	b	b
99	63	c	c	c
⋮	⋮	⋮	⋮	⋮
122	7A	z	z	z
123	7B	{	\{	\char‘\{
124	7C	|	\textbar	|
125	7D	}	\}	\char‘\}
126	7E	~	\~{}	\~{}

The following are some additional notes about the contents of Table 207:

- “"” is not available in the OT1 font encoding.
- The characters “<”, “>”, and “|” do work as expected in math mode, although they produce, respectively,

“¡”, “¿”, and “—” in text mode.[8] Hence, `$<$`, `$>$`, and `$|$` serve as a terser alternative to `\textless`, `\textgreater`, and `\textbar`. Note that for typesetting metavariables many people prefer `\textlangle` and `\textrangle` to `\textless` and `\textgreater`, i.e., “⟨*filename*⟩” instead of “<*filename*>”.

- Although “/” does not require any special treatment, LaTeX additionally defines a `\slash` command which outputs the same glyph but permits a line break afterwards. That is, “`increase/decrease`” is always typeset as a single entity while “`increase\slash{}decrease`” may be typeset with “increase/” on one line and “decrease” on the next.

- The various `\char` commands within `\texttt` are necessary only in the OT1 font encoding. In other encodings (e.g., T1), commands such as `\{`, `\}`, `\_`, and `\textbackslash` all work properly.

- `\textasciicircum` can be used instead of `\^{}`, and `\textasciitilde` can be used instead of `\~{}`. Note that `\textasciitilde` and `\~{}` produce raised, diacritic tildes. “Text” (i.e., vertically centered) tildes can be generated with either the math-mode `\sim` command (shown in Table 42 on page 21), which produces a somewhat wide “∼”, or the textcomp package’s `\texttildelow` (shown in Table 24 on page 15), which produces a vertically centered “~” in most fonts but a baseline-oriented “˜” in Computer Modern, txfonts, pxfonts, and various other fonts originating from the TeX world. If your goal is to typeset tildes in URLs or Unix filenames, your best bet is to use the url package, which has a number of nice features such as proper line-breaking of such names.

- The IBM version of ASCII characters 1 to 31 can be typeset using the ascii package. See Table 149 on page 43.

- To replace “‘” and “’” with the more computer-like (and more visibly distinct) “`” and “'” within a `verbatim` environment, use the upquote package. Outside of `verbatim`, you can use `\char18` and `\char13` to get the modified quote characters. (The former is actually a grave accent.)

Similar to Table 207, Table 208 on the next page is an amalgamation of data from other tables in this document. While Table 207 shows how to typeset the 7-bit ASCII character set, Table 208 shows the Latin 1 (Western European) character set, also known as ISO-8859-1.

The following are some additional notes about the contents of Table 208:

- A “(tc)” after a symbol name means that the textcomp package must be loaded to access that symbol. A “(T1)” means that the symbol requires the T1 font encoding. The fontenc package can change the font encoding document-wide.

- Many of the `\text...` accents can also be produced using the accent commands shown in Table 12 on page 11 plus an empty argument. For instance, `\={}` is essentially the same as `\textasciimacron`.

- The commands in the “LaTeX 2ε” columns work both in body text and within a `\texttt{...}` command (or, more generally, when `\ttfamily` is in effect).

- Microsoft® Windows® normally uses a superset of Latin 1 called “CP1252” (Code Page 1252). CP1252 adds codes in the range 128–159 (hexadecimal 80–9F), including characters such as dashes, daggers, and quotation marks. If there’s sufficient interest, a future version of the Comprehensive LaTeX Symbol List may include a CP1252 table.

While too large to incorporate into this document, a listing of ISO 8879:1986 SGML/XML character entities and their LaTeX equivalents is available from `http://www.bitjungle.com/~isoent/`. Some of the characters presented there make use of isoent, a LaTeX 2ε package (available from the same URL) that fakes some of the missing ISO glyphs using the LaTeX `picture` environment.[9]

[8] Donald Knuth didn’t think such symbols were important outside of mathematics, so he omitted them from the OT1 font encoding.

[9] isoent is not featured in this document, because it is not available from CTAN and because the faked symbols are not “true” characters; they exist in only one size, regardless of the body text’s font size.

TABLE 208: LaTeX 2ε Latin 1 Table

Dec	Hex	Char	LaTeX 2ε	
161	A1	¡	!‘	
162	A2	¢	\textcent	(tc)
163	A3	£	\pounds	
164	A4	¤	\textcurrency	(tc)
165	A5	¥	\textyen	(tc)
166	A6	¦	\textbrokenbar	(tc)
167	A7	§	\S	
168	A8	¨	\textasciidieresis	(tc)
169	A9	©	\textcopyright	
170	AA	ª	\textordfeminine	
171	AB	«	\guillemotleft	(T1)
172	AC	¬	\textlnot	(tc)
174	AE	®	\textregistered	
175	AF	¯	\textasciimacron	(tc)
176	B0	°	\textdegree	(tc)
177	B1	±	\textpm	(tc)
178	B2	²	\texttwosuperior	(tc)
179	B3	³	\textthreesuperior	(tc)
180	B4	´	\textasciiacute	(tc)
181	B5	µ	\textmu	(tc)
182	B6	¶	\P	
183	B7	·	\textperiodcentered	
184	B8	¸	\c{}	
185	B9	¹	\textonesuperior	(tc)
186	BA	º	\textordmasculine	
187	BB	»	\guillemotright	
188	BC	¼	\textonequarter	(tc)
189	BD	½	\textonehalf	(tc)
190	BE	¾	\textthreequarters	(tc)
191	BF	¿	?‘	
192	C0	À	\‘{A}	
193	C1	Á	\’{A}	
194	C2	Â	\^{A}	
195	C3	Ã	\~{A}	
196	C4	Ä	\"{A}	
197	C5	Å	\AA	
198	C6	Æ	\AE	
199	C7	Ç	\c{C}	
200	C8	È	\‘{E}	
201	C9	É	\’{E}	
202	CA	Ê	\^{E}	
203	CB	Ë	\"{E}	
204	CC	Ì	\‘{I}	
205	CD	Í	\’{I}	
206	CE	Î	\^{I}	
207	CF	Ï	\"{I}	
208	D0	Đ	\DH	(T1)

Dec	Hex	Char	LaTeX 2ε	
209	D1	Ñ	\~{N}	
210	D2	Ò	\‘{O}	
211	D3	Ó	\’{O}	
212	D4	Ô	\^{O}	
213	D5	Õ	\~{O}	
214	D6	Ö	\"{O}	
215	D7	×	\texttimes	(tc)
216	D8	Ø	\O	
217	D9	Ù	\‘{U}	
218	DA	Ú	\’{U}	
219	DB	Û	\^{U}	
220	DC	Ü	\"{U}	
221	DD	Ý	\’{Y}	
222	DE	Þ	\TH	(T1)
223	DF	ß	\ss	
224	E0	à	\‘{a}	
225	E1	á	\’{a}	
226	E2	â	\^{a}	
227	E3	ã	\~{a}	
228	E4	ä	\"{a}	
229	E5	å	\aa	
230	E6	æ	\ae	
231	E7	ç	\c{c}	
232	E8	è	\‘{e}	
233	E9	é	\’{e}	
234	EA	ê	\^{e}	
235	EB	ë	\"{e}	
236	EC	ì	\‘{ı}	
237	ED	í	\’{ı}	
238	EE	î	\^{ı}	
239	EF	ï	\"{ı}	
240	F0	ð	\dh	(T1)
241	F1	ñ	\~{n}	
242	F2	ò	\‘{o}	
243	F3	ó	\’{o}	
244	F4	ô	\^{o}	
245	F5	õ	\~{o}	
246	F6	ö	\"{o}	
247	F7	÷	\textdiv	(tc)
248	F8	ø	\o	
249	F9	ù	\‘{u}	
250	FA	ú	\’{u}	
251	FB	û	\^{u}	
252	FC	ü	\"{u}	
253	FD	ý	\’{y}	
254	FE	þ	\th	(T1)
255	FF	ÿ	\"{y}	

7.6 About this document

History David Carlisle wrote the first version of this document in October, 1994. It originally contained all of the native LaTeX symbols (Tables 27, 35, 42, 67, 84, 86, 101, 102, 108, 112, 125, and a few tables that have since been reorganized) and was designed to be nearly identical to the tables in Chapter 3 of Leslie Lamport's book [Lam86]. Even the table captions and the order of the symbols within each table matched! The $\mathcal{AMS}$ symbols (Tables 28, 43, 44, 70, 71, 87, 91, 97, and 126) and an initial Math Alphabets table (Table 135) were added thereafter. Later, Alexander Holt provided the stmaryrd tables (Tables 29, 37, 45, 73, 81, and 98).

In January, 2001, Scott Pakin took responsibility for maintaining the symbol list and has since implemented a complete overhaul of the document. The result, now called, "The Comprehensive LaTeX Symbol List", includes the following new features:

- the addition of a handful of new math alphabets, dozens of new font tables, and thousands of new symbols
- the categorization of the symbol tables into body-text symbols, mathematical symbols, science and technology symbols, dingbats, and other symbols, to provide a more user-friendly document structure
- an index, table of contents, and a frequently-requested symbol list, to help users quickly locate symbols
- symbol tables rewritten to list the symbols in alphabetical order
- appendices to provide additional information relevant to using symbols in LaTeX
- tables showing how to typeset all of the characters in the ASCII and Latin 1 font encodings

Furthermore, the internal structure of the document has been completely altered from David's original version. Most of the changes are geared towards making the document easier to extend, modify, and reformat.

Build characteristics Table 209 lists some of this document's build characteristics. Most important is the list of packages that LaTeX couldn't find, but that `symbols.tex` otherwise would have been able to take advantage of. Complete, prebuilt versions of this document are available from CTAN (`http://www.ctan.org/` or one of its many mirror sites) in the directory `tex-archive/info/symbols/comprehensive`. Table 210 shows the package date (specified in the `.sty` file with `\ProvidesPackage`) for each package that was used to build this document and that specifies a package date. Packages are not listed in any particular order in either Table 209 or 210.

Table 209: Document Characteristics

Characteristic	Value
Source file:	`symbols.tex`
Build date:	September 29, 2003
Symbols documented:	2826
Packages included:	textcomp latexsym amssymb stmaryrd euscript wasysym pifont manfnt bbding undertilde ifsym tipa wsuipa phonetic ulsy ar txfonts mathabx fclfont skak ascii dingbat skull eurosym esvect yfonts yhmath esint mathdots trsym universa upgreek overrightarrow chemarr chemarrow nath trfsigns accents nicefrac bm mathrsfs zapfchan bbold mbboard dsfont bbm
Packages omitted:	*none*

TABLE 210: Package versions used in the preparation of this document

Name	Date
textcomp	2000/08/30
latexsym	1998/08/17
amssymb	1996/11/03
stmaryrd	1994/03/03
euscript	1995/01/06
wasysym	1999/05/13
pifont	2000/01/12
manfnt	1999/07/01
bbding	1999/04/15
undertilde	2000/08/08
ifsym	2000/04/18
tipa	2001/12/31
wsuipa	1994/07/16
txfonts	2000/12/15
skak	2003/01/25
dingbat	2001/04/27
skull	2002/01/23
eurosym	1998/08/06
yfonts	2003/01/08
mathdots	2001/02/28
trsym	2000/06/25
universa	98/08/01
upgreek	2003/02/12
chemarr	2001/06/22
accents	2000/08/06
nicefrac	1998/08/04
bm	1999/07/05

References

[Ber01] Karl Berry. Fontname: Filenames for TEX fonts, June 2001. Available from `http://www.ctan.org/tex-archive/info/fontname`.

[Dow00] Michael Downes. Short math guide for LATEX, July 19, 2000. Version 1.07. Available from `http://www.ams.org/tex/short-math-guide.html`.

[Gib97] Jeremy Gibbons. Hey—it works! *TUGboat*, 18(2):75–78, June 1997. Available from `http://www.tug.org/TUGboat/Articles/tb18-2/tb55works.pdf`.

[Knu86a] Donald E. Knuth. *The TEXbook*, volume A of *Computers and Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.

[Knu86b] Donald E. Knuth. *The METAFONTbook*, volume C of *Computers and Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.

[Lam86] Leslie Lamport. *LATEX: A document preparation system*. Addison-Wesley, Reading, MA, USA, 1986.

[LAT98] LATEX3 Project Team. A new math accent. *LATEX News*. Issue 9, June 1998. Available from `http://www.ctan.org/tex-archive/macros/latex/doc/ltnews09.pdf` (also included in many TEX distributions).

[LAT00] LATEX3 Project Team. LATEX 2ε font selection, January 30, 2000. Available from `http://www.ctan.org/tex-archive/macros/latex/doc/fntguide.ps` (also included in many TEX distributions).

Index

If you're having trouble locating a symbol, try looking under "T" for "\text...". Many text-mode commands begin with that prefix. Also, accents are shown over/under a black box, e.g., "■" for "\'".

Some symbol entries appear to be listed repeatedly. This happens when multiple packages define identical (or nearly identical) glyphs with the same symbol name.[10]

[10] This occurs frequently between amssymb and mathabx, for example.

B

E

M

N

Q

R

S

T

U

V

W

X

Y

Z