Typing Documents on the UNIX System: Using Monk 0.6

S. L. Murrel
T. J. Kowalski

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

ABSTRACT

Monk is a text-formatting program that produces high-quality typeset output. This paper, printed using *monk*, describes how to produce letters, memos, release papers, and technical memoranda complete with cover sheets. We provide extensive details on the three easy steps needed to prepare a document with *monk*:

1. Use the *monksample* command to get a copy of a sample document.
2. Edit the copy, replacing the sample title, author, and the text body. Add sections, appendices, captioned figures, tables, and equations. Select the document style: character size and type, page width and length, line spacing, and number of columns.
3. Format the document using the *monk* command, sending the output to a 5620 or 630MTG terminal or a printer. *Monk* will automatically invoke the table, picture, equation, graph, and reference pre-processors in the correct order.

1. Introduction

Preparing any document involves positioning words and figures on each page. Using a typewriter, the margins, line spacing, and tab positions are selected by moving levers. Text is positioned by moving the sheet of paper left, right, up or down. Using a computer, the text is interspersed with commands that tell a formatting program what to do.

Monk is a new package of commands for producing AT&T letters, memos, technical memoranda, and their cover sheets. This paper will enable you to prepare documents using *monk* [9]-[11]. You need to know how to log onto a computer and how to use an editor. If you don't, see the UNIX Text Editing and Phototypesetting Starter Package [12].

2. Mixing Text and Commands

Using any text editor, type normally. However, instead of skipping a line before paragraphs type `|paragraph` or `|p`. This command will produce a blank line. To indent, use `|p(i)` or `|paragraph(indented)`; to number, use `|p(n)` or `|paragraph(numbered)`.

There are other commands that let you select document style and character type (e.g., double spacing and italics) and include various sections and displays (e.g., chapters and tables).

3. Command Scope

Using *monk*, the sentence ''Italicize the words in |italics(parentheses)'' produces ''Italicize the words in *parentheses*.'' All *monk* commands begin with the character '|'. The command `|italics` selects the operation and the parentheses delimit the words to be italicized. There are seven pairs of delimiters:

`(...)`, `[...]`, `<...>`, `{...}`, `"..."`, `'...'`, `‘...’`

Always choose delimiters so that the closing delimiter does not appear in the text. ''Italicize the equation |i(a = (b + c) / number)'' produces ''Italicize the equation *a = (b + c* / number).'' The closing delimiter ')' is found after the letter 'c', so that the rest of the equation is not italicized. The input ''Italicize the equation |i<a = (b + c) / number>'' will correctly italicize the entire equation, *a = (b + c) / number*. In contrast, when *monk* commands are inside other *monk* commands, the delimiter pairs are not ambiguous. ''Italicize and shrink this |small(|i(word) but only shrink the rest)'' produces ''Italicize and shrink this *word* but only shrink the rest.''

There is a second form of delimiter — when a command operates on a long text segment, the commands `|begin` and `|end` are used instead of brackets or quotes. The text fragment

```
|begin(italics)
```

```
Four score and seven years ago our fathers
brought forth to this continent a new
nation, conceived in liberty and ...
|end(italics)
```

appears in the final document as *Four score and seven years ago our fathers brought forth to this continent a new nation, conceived in liberty and ...* The commands `|begin` and `|end` delimit the text operated on by the command `|italics`.

4. Document Commands

This section describes the commands that are specific to formatting four types of documents: letters, AT&T Bell Laboratories memoranda, technical memoranda, and release papers.

4.1 Typesetting a Letter

For a letter, the input should begin as follows:

```
|make(letter)
|begin(ret_address)
|s(AT&T) Bell Labs 2C-501
Murray Hill, N. J. 07974
|end(ret_address)
```

The first line specifies that the standard format for a letter should be used. Then the return address, labeled `ret_address`, is delimited by `|begin` and `|end` commands. The formatter skips a couple of lines, indents, single spaces the address itself, skips more lines, and then returns to the left hand margin producing:

AT&T Bell Labs 2C-501
Murray Hill, N. J. 07974

The return address itself is not modified. If the city, state, and zip code are input as separate lines, they will appear that way in the final output:

AT&T Bell Labs 2C-501
Murray Hill
N. J.
07974

A letter consists of a sequence of standard groups: a return address, date, recipient's address, greeting, body, closing, and signature. Accordingly, the following prototype letter is available.

```
|make(letter)

|begin(ret_address)
|s(AT&T) Bell Laboratories 2C-501
Murray Hill, New Jersey 07974
|end(ret_address)

|date(April 1, 1985)

|begin(address)
G. H. Reiber
|s(AT&T) Communications
Room 3C-110
Bedminster, |s(NJ) 07921
|end(address)

|greeting(George Reiber:)

|begin(body)
Yes, I would like to hire a painter for the
night of Monday, September 16, 1985.
Thank you very much.
|end(body)

|closing(Sincerely,)

|signature(G. R. Emlin)
```

To copy this letter into a file named `myfile` in your current directory, type the UNIX command

```
monksample letter > myfile
```

Using this prototype, it is not necessary for you to remember that the label for return address is `ret_address`. You need only recognize the English-like descriptions and replace the sample text as appropriate. Although any text editor will suffice, the large display of a visual editor allows you to glance at the file and spot sample text.

It is convenient to tailor the samples by entering your address, closing, and signature and then saving the modified file. This new file can then be used for all your letters simply by adding an appropriate greeting, the recipient's address, and the body. If you prepare letters for more than one person, you might want to maintain a directory of tailored prototypes rather than begin with the standard *monk* sample.

```
|make(letter)

|begin(ret_address)
|s(AT&T) Bell Laboratories 2C-524
Murray Hill, New Jersey 07974
|end(ret_address)

|date(April 1, 1985)

|begin(address)
G. H. Reiber
|s(AT&T) Communications
Room 3C-110
Bedminster, |s(NJ) 07921
|end(address)

|greeting(To Whom It May Concern:)

|begin(body)
Thank you very much.
|end(body)
```

```
|closing(Sincerely,)
|signature(Sam)
```

When installed, *monk* is configured to use local printers as standard output devices.† You will need to talk to someone using or maintaining your computer system to find out what and where your local standard printer is.

To run off this memo on the standard printing device, type `monk myfile`. To send your output to a printing device other than the standard, use the `–S`*dest* and `–T`*font* options. To preview postscript output on a AT&T 5620 or 630MTG terminal, type `monk –Tpost –Sproof myfile`. Devices and fonts will vary with your location, but commonly available types include:

`–S`*dest*	**Sends Output To Device** *dest*
`–Sapsend`	Linotronic 200P, see *apsend*(1)
`–Sd202`	Mergenthaler Linotronic 202, see *d202*(1)
`–Si300`	Spooling for EXPTOOLS, see *i300*(1)
`–Slp`	Spooling mechanism, see *lp*(1) (default)
`–Slpr`	Spooling at Allentown, see *lpr*(1)
`–Sproof`	5620 or 630MTG terminal, see *proof*(9.1)
`–Sprt`	Spooling at computer centers, see *prt*(1)
`–Sthink`	Thinkjet, see *thinkblt*(9.1)
`–S–`	Standard output

`–T`*fonts*	**Selects Typesetter Or Terminal** *fonts*
`–T202`	Mergenthaler Linotronic 202
`–Taps`	Autologic APS-5
`–Ti10`	Imagen 10 printer
`–Ti300`	Imagen 300 printer
`–Tim300`	Imagen 300 printer at Liberty Corner
`–Tpost`	PostScript® printer (default)
`–Tthink`	Think jet printer

If TYPESETTER is set and exported, then it determines the typesetter or terminal fonts used in preparing the output. Be careful to type `monk –Tpost –Sapsend myfile` or `monk –Taps –Sprt myfile` **only** when you intend to send final copy to the high-quality, high-priced phototypesetter.

When *monk* is run, the individualized prototype letter will appear as a full size version of the page below. This is an excerpt of the sample's output, scaled and included using PostScript.

† Specify the printer that you have permission to use by defining LPDEST in your *.profile*, e.g., `LPDEST=crete; export LPDEST`.

AT&T Bell Laboratories 2C-524
Murray Hill, New Jersey 07974

April 1, 1985

G. H. Reiber
AT&T Communications
Room 3C-110
Bedminster, NJ 07921

To Whom It May Concern:

Thank you very much.

Sincerely,

Sam

4.2 Typesetting a Memorandum

Preparing a memorandum is similar to preparing a letter. Memoranda require a title, more information about the author, and the document itself. The information you provide is automatically arranged at the top of the page.

Type

```
monksample memo > myfile
```

to get the following prototype:

```
|make(memo)
|author(name "W. H. MacWilliams",
   initials WHM, location WH, room 2E-221,
   extension x511, department 1344)
|document(number 1344-561210-09,
   file_case 25952, work_program 311401-0199)
|date(December 10, 1956)
|title(|s(MTS) Employment: Dr. J. L. Flanagan)
|titlebox
F. D. Leamer:
|p
This is to propose an offer of regular
|s(MTS) employment to Dr. J. L. Flanagan
commencing on February 18, 1957.
|signature(WH-1344-WHM-hhs)
|notation[Copy (with att.) to:
E. E. David Jr.
A. G. Jensen]
|approved(J. R. Pierce)
```

The `|make` command asks *monk* to format the document as a memo. The information about each

author is given in a separate `|author` command: name, initials, location, department, extension, and room number. Each instance of `|document` gives the document information for the corresponding author. Author, document, title, and date may appear in any order, but must precede the command `|titlebox`, which organizes, formats, and outputs all this information.

After the titlebox comes the body of the memo, which consists of paragraphs separated by `|p`, simple font changes, and figures. After the body comes the signature, any necessary notations, and approvals. This sample produces the following memo:

AT&T

subject: **MTS Employment: Dr. J. L. Flanagan**

Work Program 311401-0199 File Case 25952

date: **December 10, 1956**

from: **W. H. MacWilliams**
WH 1344
2E-221 x511

1344-561210-09

F. D. Leamer:

This is to propose an offer of regular MTS employment to Dr. J. L. Flanagan commencing on February 18, 1957.

WH-1344-WHM-hhs

W. H. MacWilliams

Copy (with att.) to:
E. E. David Jr.
A. G. Jensen

Approval:

J. R. Pierce

To omit information, simply delete the appropriate *monk* command. Delete the entire line to omit notations or approvals. To omit an author's location, omit the relevant labels and associated numbers, e.g., `|author(name "W. H. MacWilliams", initials WHM, department 1344)`.

4.3 Typesetting a Technical Memorandum

This section describes how to prepare a technical memorandum. The example below is the skeleton of the file that produced the previous *tm* version of this document, excluding cover sheet information. This prototype consists of the declaration of the document type, document numbers, date, authors, title, abstract, document body, appendices, and references. It can be copied with the command `monksample tm > myfile`.

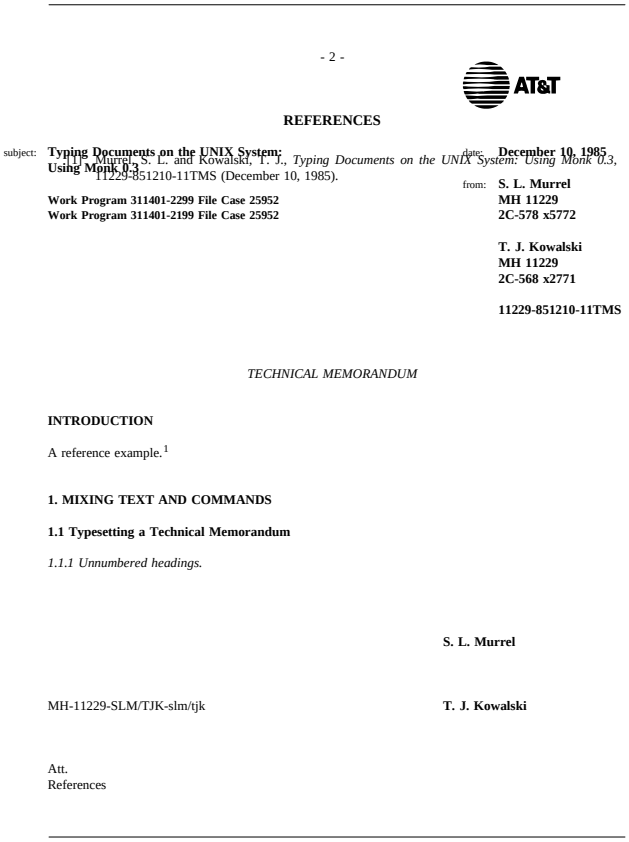
```
|make(tm)

|author(name "S. L. Murrel", initials SLM,
        location MH, department 11229,
        extension x5772, room 2C-578)
|document(number 11229-851210-11TMS,
        file_case 25952,
        work_program 311401-2299)
|author(name "T. J. Kowalski", initials TJK,
        location MH, department 11229,
        extension x2771, room 2C-568)
|document(number 11229-851210-11TMS,
        file_case 25952,
        work_program 311401-2199)

|date(December 10, 1985)

|title(Typing Documents on the UNIX System:
Using Monk 0.3)

|begin(abstract)
|i(Monk) is a text-formatting program that
produces high-quality printed text.
|end(abstract)

|titlebox

|unnumbered_section(INTRODUCTION)

A reference example.|reference(
Murrel Kowalski Using Monk 0.3 1985)

|section(MIXING TEXT AND COMMANDS)

|subsection(Typesetting a Technical Memorandum)

|subsubsection(Unnumbered headings.)

|signature(MH-11229-SLM/TJK-slm/tjk)

|notation(Att.
References)

|other

|reference_placement
```

The first line should specify the document type. After that the order is flexible, with the one restriction that all the component information be available when larger units, such as the titlebox, are printed. Thus the author, date, and document information must be specified before the command `|titlebox`, because it uses that information to produce the standard header at the top of technical memoranda. Note that the same `|titlebox` command produces different output depending on the document type specified.

The title is not filled, so that newlines entered by the author are preserved: here `Using Monk 0.3` appears as the second line of the title.† This is the first

† In unfilled text, input lines are reproduced unchanged in the output; in filled text, words from successive input lines may be combined to ensure maximum length lines in the output.

page produced by the *tm* input sample, scaled to fit into the column. The second page of the output provides the formatted reference list.



- 2 -

AT&T

REFERENCES

subject: **Typing Documents on the UNIX System: Using Monk 0.3**

date: **December 10, 1985**

from: **S. L. Murrel**
MH 11229
2C-578 x5772

T. J. Kowalski
MH 11229
2C-568 x2771

11229-851210-11TMS

Work Program 311401-2299 File Case 25952
Work Program 311401-2199 File Case 25952

TECHNICAL MEMORANDUM

INTRODUCTION

A reference example.[1]

1. MIXING TEXT AND COMMANDS

1.1 Typesetting a Technical Memorandum

1.1.1 Unnumbered headings.

S. L. Murrel

MH-11229-SLM/TJK-slm/tjk

T. J. Kowalski

Att.
References

[1] Murrel, S. L. and Kowalski, T. J., *Typing Documents on the UNIX System: Using Monk 0.3*, 11229-851210-11TMS (December 10, 1985).

The commands `|author` and `|document` belong to a type we have not previously discussed. They allow you to specify a collection of information about a given item. The first instance of `|author` specifies the name, initials, location, department, extension, and room number of the first author. The first instance of `|document` gives the document information for the first author. Authors and documents are linked as they appear, so that

```
|author(name "S. L. Murrel", initials SLM,
   location MH, department 11229)
|author(name "T. J. Kowalski", location MH,
   initials TJK, department 11229)
|document(number 11229-851210-11TMS,
   file_case 25952, work_program 311401-2299)
|document(number 11229-851210-11TMS,
   file_case 25952, work_program 311401-2199)
```

is equivalent to the ordering given in the *tm* above. Note that in these examples only part of the author information is included: the room and extension numbers were omitted. Any information can be omitted simply by deleting the corresponding *monk* labels, e.g., `room` and `extension`. Monk will try to format the information given gracefully.

Additional information is needed to produce a standard AT&T cover sheet:

```
|keywords(Word Processing,
Text Processing,
Formatting)

|mercury(cmp, elc)

|att(is_release yes)

|distribute_complete_memo(Exec Directors 112
Directors 112
Department Heads 1122)

|distribute_cover_sheet(A. A. Penzias
1122 MTS)

|cover_sheet
```

Again, this information can appear anywhere in the document as long as it precedes the command `|cover_sheet`. Following local convention, keywords are filled, while the distribution lists are not. Although the title was not filled in the titlebox, it is filled on the cover sheet because of space limitations.

In producing the cover sheet, *monk* automatically calculates the numbers of tables, figures, references, as well as text pages, other pages, and total pages. Use the command `|other` to mark the end of the text body and the beginning of attachments. *Monk* needs it to correctly distinguish text from other pages. If there are any pages that are not included in this input file, you can add to the counts using the command `|add_totals`. For example, if there is an additional attachment page that is not a table or figure, then the command `|add_totals(other 1)` suffices. You should never have to use the complete form of the command: `|add_totals(text 0, tables 0, figures 1, references 11, other 2)`. This command means that you are adding 2 additional pages by hand, which include one figure and eleven references. The counts of text pages and tables are untouched.

There are six commands specific to the AT&T Bell Labs cover sheet that provide distribution information. Use `|mercury` to specify the company-wide announcements that should include the abstract of this document: `chm`, `cmm`, `cmp`, `elc`, `lfs`, `man`, `mas`, `mkt`, and `phy`. Use the *monk* command `|distribute_complete_memo` to list individuals who should receive the complete memo and `|distribute_cover_sheet` to list those who should receive only the cover sheet. The command `|att` gives relevant AT&T release information:

`is_release` marks the *yes* or *no* box as specified by its argument. `|director_name` identifies the director of the first author. Finally, the command `|cover_sheet` prints the following cover sheet. Note that this sample produces four pages: one page of text, one page of references, and two pages of cover sheet.

AT&T Bell Laboratories **Document Cover Sheet for Technical Memorandum**

Title: Typing Documents on the UNIX System: Using Monk 0.3 **Author's Date:** December 10, 1985

Authors	Location	Ext.	Dept.
S. L. Murrel	MH 2C-578	x5772	11229
T. J. Kowalski	MH 2C-568	x2771	11229

Document Nos.	Filing Case Nos.	Project Nos.
11229-851210-11TMS	25952	311401-2299
11229-851210-11TMS	25952	311401-2199

Keywords:

Word Processing, Text Processing, Formatting

MERCURY Announcement Bulletin Sections:

CMP - Computing ELC - Electronics

Abstract:

Monk is a text-formatting program that produces high-quality printed text.

Total Pages (including document cover sheet): 4 **Mailing Label**

AT&T — PROPRIETARY
Use pursuant to Company Instructions

4.4 Typesetting a Release Paper

There is a *monk* sample for release papers, which can be modified in the same way as the documents already discussed. However, you may often want to convert a technical memorandum into a release paper. To do so, change the first line from `|make(tm)` to `|make(rp)` and then remove or comment out (see Section 5.1) the information specific to the *tm* style:

- `other`
- `add_totals`
- `mercury`
- `distribute_cover_sheet`
- `distribute_complete_memo`
- `cover_sheet`
- `att`
- `signature`
- `director_name`

5. Content Formatting Commands

For any document type, it is possible to format section titles, figure captions, footnotes, and two-column output. It is possible to store page numbers, section numbers or figure numbers for later use. An interface is provided to existing routines that produce tabular output, equations, citations, and reference lists. This section describes the relevant commands.

5.1 Comments

Comments are valuable notations by and for the person preparing a document. They can be used to temporarily omit a section, to mark a section that needs work or information, and to retain an alternative word or phrase. Anything that appears within the delimiters of a `|comment` is not processed as part of the document: *monk* commands are ignored, and text does not appear in the final output.

5.2 Spacing

You can start a new page or leave vertical blank space using these commands.

`|new_page`	begins a new page
`|new_column`	begins a new column
`|blank_space`	leaves a blank line
`|blank_space(3)`	leaves 3 blank lines
`|blank_space(1.5i)`	leaves 1.5 inches
`|blank_space(-1)`	moves back one line

5.3 Positioning Text: Indenting and Centering

Indenting and centering are the most common ways of positioning text. The input text

```
|indent(begins a new line and indents
the left hand margin as long as it applies
for a single line, a group of lines,
or an entire paper.
|indent<Multiple commands will
indent progressively.>)
```

produces the following output:

> begins a new line and indents the left hand margin as long as it applies for a single line, a group of lines, or an entire paper.
>
> > Multiple commands will indent progressively.

The command `|center` begins a new line and centers each line of text within its delimiters. The

input, taken from Monty Python,

```
|begin(center)
I'm a lumberjack and I'm okay,
I sleep all night and I work all day.
I cut down trees, I eat my lunch,
I go to the lavat'ry,
On Wednesdays I go shopping and have
buttered scones for tea.
|end(center)
```

produces

I'm a lumberjack and I'm okay,
I sleep all night and I work all day.
I cut down trees, I eat my lunch,
I go to the lavat'ry,
On Wednesdays I go shopping and have
buttered scones for tea.

In contrast, the command `|center_block` centers the entire figure as a block. All the lines are left-justified and the figure is centered based on the width of the longest line. It produces

I'm a lumberjack and I'm okay,
I sleep all night and I work all day.
I cut down trees, I eat my lunch,
I go to the lavat'ry,
On Wednesdays I go shopping and have
buttered scones for tea.

Multi-line quotations are appropriately indented and offset by the command `|quotation`.

> Four score and seven years ago our fathers brought forth to this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal.

5.4 Document Styles

By default, all documents are single-spaced and single column. Page width is 6 inches; length is 11 inches. The font family is *times*; the font size is 10p on a vertical spacing of 12p. Paragraphs skip a line and begin the text flush with the left margin.

You can change any of these defaults using the `|style` command. The command `|style( page_width 7i)` sets the page width to 7 inches. To get the maximum page width on most output devices, type: `|style(page_offset 0, page_width 8.5i)`. The `page_length` and `font_size` can be set in a similar manner.

`|style(font_size 14)` changes the default font to size 14 to print out your papers for your grandparents or to prepare invitations to a party. `|big` or `|small` will still provide one size bigger or smaller, respectively.

Table 1. The alphabet in common type families

times	roman	abcdefghijklm...uvwxyz
	italics	*abcdefghijklm...uvwxyz*
	bold	**abcdefghijklm...uvwxyz**
palatino	roman	abcdefghijklm...uvwxyz
	italics	*abcdefghijklm...uvwxyz*
	bold	**abcdefghijklm...uvwxyz**
helvetica	roman	abcdefghijklm...uvwxyz
	italics	*abcdefghijklm...uvwxyz*
	bold	**abcdefghijklm...uvwxyz**

Font families provide complementary styles for each of the four fonts: bold, italics, roman, and cw. The standard font family is times. Alternative font families include palatino, helvetica, bembo, optima, souvenir, and cw, but not all families are available on all output devices. Some examples are shown in Table 1. To select helvetica, type `|style( font_family helvetica)`.

`|style(double_space)` double spaces the text, while `|style( single_space)` restores single spacing. You can change the paragraph style throughout the rest of your document by typing the command `|style(paragraph_style indented)` or `|style(paragraph_style numbered)`, after which the simple command `|p` will produce indented or numbered paragraphs. To indent or number a single paragraph, use `|p(i)` or `|p(n)`, respectively. To restore the original style paragraph, type `|style( paragraph_style flushed)`.

The command `|style(one_column)` begins formatting text in one column. The command `|style(two_column)` restores two column formatting.

5.5 Fonts

Monk currently provides abbreviated commands for four type faces and two type sizes. `|bold` or `|b` makes the enclosed text **bold**. `|italics` or `|i` produces *italicized text*, while `|roman` or `|r` restores the roman character set. The command `|constant_width` or `|cw selects a font that resembles low quality line printer output and is intended only`

`for examples of such output`.

When fonts are embedded, the innermost dominates. For example, `|italics<Italics flourish around the |roman[roman] text>` produces *Italics flourish around the* roman *text*. If an output device is not able to italicize, then it will underline. Because italics look better, it is always preferable to request them so that the document will look its best on the available printing device. However, it is possible to specify underlining using `|underline` or `|u`.

`|big` makes the font size bigger, while `|small` makes the font size smaller. These can be nested, so that `|big(|big(this is big))` produces this is big. However, `|big` and `|small` are designed for in-line font changes; they do not change the vertical spacing. To change the size of a block of text, use the `|style` command (see Section 5.4).

5.6 Special characters

Monk provides a long list of special characters, the appearance of which is ultimately determined by the output device. With the command `|sp`, the user can print out the *monk* command character `|`, Greek letters, and mathematical symbols.

Here is a partial list of the fancy symbols that appear in Appendix B: / − \ © ® T^M | ¦ † § ☞ ☜ - _ ‘ ’ ` ´ ¨ α β γ δ ε ± ≥ = ≠ ≃ ~ × ÷ ∝ ∫ ∂ ¬ | ⌉ □ ∇ ∅ ∈ ⊇ ∩ → ↑ ↓ ¢ ¼ ¾ μ ν ξ o π ff ffi ffl TEX LaTeX.

`|sp` can also be used to control spacing and carriage movement precisely. There are spaces of five different widths, shown below opposite the commands that produce them.

```
|sp{space}                ‘ ’
|sp{digit space}          ‘ ’
|sp{narrow space}         ‘’
|sp{half narrow space}    ‘’
|sp{zero width space}     ‘’
```

The carriage can be moved up and down: `|sp{up}` moves up, while `|sp{reverse}` moves up by twice as much. The command `|sp{down}` moves down.

`|sp{break and spread}` pads the current line according to the document style and breaks to the next line. To force hyphenation at a particular position in a word, use `|sp{hyphenation}`. `|sp{tab}` generates horizontal motion to the next tab stop; `|sp{leader}` generates a string of periods to the next tab stop.

5.7 Headings

5.7.1 Numbered headings. `|chapter` starts a new page; leaves a blank space; puts ‘‘Chapter *n*’’ against the right hand margin; skips more space; puts the title against the right hand margin; and skips more space before beginning the text body. No chapters appear in this document. The commands to declare the current section, subsection, and subsubsection include:

```
|section(Content Formatting Commands)
|subsection(Headings)
|subsubsection(Numbered headings.)
```

The first appendix at the end was produced by `|appendix(List of Commands)`.

i) This numbered paragraph is produced by the command `|paragraph(numbered)`, which may be abbreviated as `|paragraph(n)` or `|p(n)`.

5.7.2 Unnumbered headings. The analogous unnumbered headings produce the same spacing but omit heading names, e.g. chapter, and numerical labels. Chapter titles remain right justified, and appendix titles are still centered; but the various section titles are now flush with the left margin. `|unnumbered_chapter` is useful for preface and acknowledgement pages. Available commands include:

```
|unnumbered_chapter(Preface)
|unnumbered_section(SECTION)
|unnumbered_subsection(Subsection)
|unnumbered_subsubsection(Subsubsection.)
|unnumbered_appendix(APPENDIX)
```

There are flushed and indented unnumbered paragraphs. The indented paragraphs are used throughout this paper, except for this paragraph, which is produced using the command `|p(f)`. The commands to skip a line and begin the text flush with the left margin include:

```
|paragraph(f)
|p(f)
|paragraph(flushed)
|p(flushed)
```

The commands to indent the first line of the paragraph and produce spacing identical to the flushed paragraph include:

```
|paragraph(i)
```

```
|p(i)
|paragraph(indented)
|p(indented)
```

5.7.3 Paragraphs. The command `|paragraph` or its abbreviation `|p` is used throughout this paper to produce the default paragraph style. In this paper the normal default, `flushed`, was changed through use of the `|style` command to `indented` (see Section 5.4).

5.8 Setting page, heading and figure numbers

Page, heading, figure, and table numbers can be changed by the authors. Set the current page counter to three using the command `|set_counter(page 3)`. At the next page boundary the page number will be incremented to four. The command `|set_counter( table 12)` sets the current table number to 12. Multiple counters can be set with a single command: `|set_counter(page 10, chapter 4, section 3, subsection 1)`. Available commands include:

```
|set_counter(page 3)
|set_counter(chapter 4)
|set_counter(section 1)
|set_counter(subsection 9)
|set_counter(subsubsection 3)
|set_counter(paragraph 0)
|set_counter(appendix 3)
|set_counter(footnote 17)
|set_counter(note 122)
|set_counter(figure 8)
|set_counter(table 0)
```

Care must be taken in positioning all these commands: if you want to change a section number, then you must position the `|set_counter` command before the section itself so that the numbered title will be correct. To number the next section four, you must type `|set_counter(section 3)` before the `|section` command, because the section number will be incremented when the `|section` command itself is read. Similarly, if you want the page number in headers and footers to display a new value, then the value must be changed before the break to the new page occurs. To set the next page to page number three, the command `|set_counter(page 2)` must appear on the current page.

Once a value is set, the computer will use that number and increment it appropriately; but if it is set wrong, it will remain wrong. If you change counters, you are responsible for insuring consistency and correctness.

5.9 Setting page headers and footers

Both page headers and footers can be set using the `|set_string` command as follows:

```
|set_string(page_headers "'left'center'right'")
|set_string(even_headers "'left'center'right'")
|set_string(odd_headers "'left'center'right'")
|set_string(page_footers "'left'center'right'")
|set_string(even_footers "'left'center'right'")
|set_string(odd_footers "'left'center'right'")
```

`Page_headers`, `even_headers`, and `odd_headers` set the headers on all pages, even pages, and odd pages, respectively. Page headers appear above the even and odd headers. All headers consist of three strings, any of which may be blank or long enough to overlap. Any character, can be used to delimit the strings. The strings *left*, *center*, and *right* are left-justified, centered, and right-justified, respectively. Parallel commands serve to specify page footers.

5.10 Lists

There are nine types of lists, which differ only in the labeling of list items:

```
|number_list
|A_list
|a_list
|I_list
|i_list
|p_number_list
|p_a_list
|dash_list
|bullet_list
```

A simple numbered list is created by the following commands:

```
|begin(number_list)
|item first item
|item second item is on two lines in the
input, but will be filled in the output
|item third
|end(number_list)
```

`|A_list` items are marked with capital letters; `|a_list` with lower-case letters followed by ')'. `|I_list` items are marked with upper-case roman numerals; `|i_list` with lower-case roman numerals followed by ')'. Number and lower-case letter labels are enclosed in parentheses in `|p_number_list` and `|p_a_list`, respectively. Bullets or dashes label the items in a `|bullet_list` or a `|dash_list`, respectively.

Multi-level lists of up to six levels can be built

by putting lists inside lists. A standard outline format of six levels consists of an outer `|I_list`, an `|A_list`, a `|number_list`, an `|a_list`, an inner `|number_list`, and a `|p_a_list`. The following produces, in outline format, a description of two kingdoms.

```
|begin(I_list)
|item Animal Kingdom
|begin(A_list)
 |item Arthropods
|begin(number_list)
  |item Trilobites
  |item Insects
|end(number_list)
 |item Chordates
|begin(number_list)
  |item Vertebrates
|begin(a_list)
   |item Birds
   |item Reptiles
   |item Mammals
|begin(number_list)
    |item Marsupials
    |item Monotremes
|begin(p_a_list)
     |item platypus
     |item echidna
|end(p_a_list)
    |item Placental mammals
|begin(p_a_list)
     |item Cetacea
     |item Primates
|end(p_a_list)
|end(number_list)
|end(a_list)
  |item Notocords
|end(number_list)
 |item Mollusks
|end(A_list)
|item Plant Kingdom
|end(I_list)
```

I. Animal Kingdom
   A. Arthropods
      1. Trilobites
      2. Insects
   B. Chordates
      1. Vertebrates
         a) Birds
         b) Reptiles
         c) Mammals
            1. Marsupials
            2. Monotremes
               (a) platypus
               (b) echidna
            3. Placental mammals
               (a) Cetacea
               (b) Primates
      2. Notocords
   C. Mollusks
II. Plant Kingdom

5.11 Inserting other files

Other files may be inserted using the command `|insert`. These files will also be processed by *monk* and may in turn insert other files. Loops will indeed cause disaster. A file that will not be processed may also be included using the `|source` command.

5.12 Figures

The lumberjack poem above was formatted as a figure. Figures are printed with blank space above and below and can be split across a page boundary. Lines are not filled. If you want some lines to be filled, put them within the delimiters of the command `|text`. `|figure_caption` centers and labels the caption. Given

```
|begin(figure)
Like the lumberjack poem above,
this text is within a |cw<|sp(|)figure>
command.
|end(figure)
|figure_caption(This is a figure caption).
```

monk produces

Like the lumberjack poem above,
this text is within a `|figure`
command.

Figure 1. This is a figure caption

To create a non-filled display without spacing above and below, use the command `|verbatim`. To indent either a figure or a verbatim display, simply enclose it within an `|indent` command:

```
|begin(here)
|begin(indent)
|begin(figure)
This is a short figure
|end(figure)
|end(indent)
|figure_caption(with a short caption)
|end(here)
```

The command `|indent` can appear either inside or around the `|figure` command. However, because the caption is centered, it should not be indented.

The `|here` command keeps all the text within its delimiters together on one page. It is important to include both the figure proper and its caption within the delimiters so that the caption is never orphaned. If

there is not enough room on the current page, `|here` leaves the rest of the page blank and skips to the next page or next column to print the block. It is useful after comments such as ‘‘An example follows:’’, because it guarantees the relative position of the comment and the block.

Another command, `|around`, also keeps the enclosed text together. If it cannot print the block on the current page, however, it continues printing text, postponing the block until there is enough room for it. Usually the block will be positioned at the top of the next page; however, if there are multiple large floating blocks that cannot fit together on that page, then they will appear in order as soon as possible. You can use the command `|around_placement` to force all outstanding blocks to be printed at the end of a section or chapter.

5.13 Equations

Monk uses *eqn* [6] to format equations. The *monk* command `|equation` delimits an equation offset from the text. Note that *monk* commands may appear within the line; they do not need to be at the beginning of a line.

```
|begin(center)
|equation_number[(1a)]
|begin(equation)x = 2 pi int
sin ( omega t ) dt
|end(equation)
|end(center)
```

When formatted the above input produces:

$$x = 2\pi \int \sin(\omega t)\, dt \qquad (1a)$$

The same output is produced by the short *monk* form: `|equation[x = 2 pi int sin ( omega t ) dt]`. Captions are specified using `|equation_caption`.

The equation itself is explained on page 2 of the Kernighan and Cherry memo.[1] To write in-line equations, delimit the equation using `|e`. The input `|e<x = 2 pi int sin ( omega t ) dt>` produces the equation $x = 2\pi \int \sin(\omega t)\, dt$ without offsetting it.

1. Remember that the *monk* command `|equation(...)` replaces all instances of `.EQ` and `.EN` and that `|e(...)` replaces the `$...$` delimiters for in-line equations.

5.14 Tables

Monk does not provide any new capabilities for formatting tables; it uses *tbl* [2]. It provides *monk*-style commands for positioning both the table and its caption and automatically invokes the standard *tbl* formatting program. The following input formats the table in the file *examples/example.table*, which appears as Appendix C.1.[2] The table could be typed into the file directly; here it is put into another file just to simplify and condense the *monk* example.

```
|begin(here)
|table_caption(Example of table)
|begin(table)
|insert(examples/example.table)
|end(table)
|end(here)
```

The command `|here` guarantees that the caption will not be separated from the table if it happens to hit a page boundary. The remainder of the page would be blank, and the table with caption would be printed on the next page or in the next column for multi-column documents. Together *monk* and *tbl* produce

Table 2. Example of table

Major New York Bridges		
Bridge	Designer	Length
Brooklyn	J. A. Roebling	1595
Manhattan	G. Lindenthal	1470
Williamsburg	L. L. Buck	1600
Queensborough	Palmer & Hornbostel	1182
Triborough	O. H. Ammann	1380
		383
Bronx Whitestone	O. H. Ammann	2300
Throgs Neck	O. H. Ammann	1800
George Washington	O. H. Ammann	3500

If you have a 5620 or a 630MTG terminal, the interactive table editor *vtbl* can be used to prepare *tbl* input [8]. Both the table structure and its text contents can be edited using *vtbl*.

2. The formatting commands `.TS` and `.TE` need to be omitted from the sample file.

5.15 Computer Output

The command `|computeroutput` produces a figure using a constant width type that resembles low quality line printer output.

```
Program: SINDSCAL-symmetric indscal

*********************************
parameters
 dim  irdata  itmax   irn
  2      2    200     9342
no. of matrices=5  no. of stim.=9
*********************************

 correlations    vaf       loss
 y(data)&yhat   (r**2)  (y-yhat)**2
   .161169     .025975    .974025
   .650655     .423352    .605140
```

Figure 2. Example of computer output

`|computeroutput_caption (Example of computer output)` provides the associated caption. Currently these captions are the same as figure captions.

5.16 Pictures

Monk commands also provide access to *pic*, a graphics language for typesetting [5]. Borrowing an example from the *PIC User Manual*,[3] the *monk* input

```
|begin(here)
|begin(picture)
|insert(examples/example.pic1)
|end(picture)
|picture_caption(Example of picture)
|end(here)
```

produces

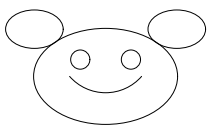


Figure 3. Example of picture

If you have a 5620 or a 630MTG terminal, the interactive drawing program *cip* can be used to prepare *pic* input [1].

3. Remember that the *monk* command `|picture(...)` replaces all instances of PS and PE.

Another example from Kernighan's *PIC Revised User Manual* illustrates the use of an in-line equation embedded in a picture.[4]

```
|begin(here)
|begin(picture)
arrow
box "|e[{H( omega )} over {1 - H( omega )}]"
arrow
|end(picture)
|picture_caption(Picture with equation)
|end(here)
```

produces the following beauty

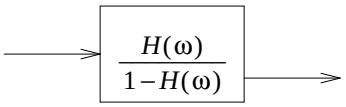


Figure 4. Picture with equation

5.17 Footnotes

A numbered footnote is generated by the command `|footnote` appended to the running text.[5] The text `|footnote(This is yet another numbered footnote)` produces numbered footnotes.[6] Analogously, `|dagnote` and `|starnote` produce daggered footnotes† and starred footnotes.* Footnote types can be mixed, as in this section.

5.18 Citations

Monk uses the program *cite* to handle citations. The command `|save(tag)` remembers the last important numbered item as the label *tag*. It saves the number of the last figure, picture, equation, table, section, subsection, subsubsection or appendix. At this point the command `|save(this_section)` stores the number of the current subsection as `this_section`. The *monk* command `|remember(this_section)` is replaced by that stored value wherever it appears in the text: `This`

4. In this example, the equation within the delimiters must be placed within the `|e` command.

5. This is a numbered footnote

6. This is yet another numbered footnote

† This is a daggered footnote

* This is a starred footnote

`section is |remember(this_section)` appears in the final output as This section is 5.18. It is crucial to understand that `|save` stores the most recent counter, whatever its type. If a numbered figure appears between the section header and the `|save` command, then the number saved will be that figure number. The tag name itself has no particular meaning. Position is what counts.

Page numbers can be saved using the command `|savepage. |savepage(tag)` saves the current page number as the label `tag`. Again, `|remember(tag)` substitutes the number associated with `tag`. The text `the current page is |savepage(foobar) |remember(foobar)` generates `the current page is 251`.

If all citations are saved before they are used, then the user need only enter the commands shown above in the *monk* input text. If, however, the input uses a citation before it really is saved, as in the example below, then there is more to do.

```
All commands are listed in appendix
|remember(appendix_commands).
…
|appendix(LIST OF COMMANDS)
|save(appendix_commands)
```

Forward referencing depends on a file named `.cite`, which is generated each time the input is formatted. The UNIX command

```
cite –s .cite > myfile.cite
```

reports any redefinitions or unused definitions and produces a consistent set of citations. The output file `myfile.cite` needs to be inserted in the *monk* input before the first citation. Do so using the command `|insert(myfile.cite)`.

5.19 References

Monk uses *prefer* to extract references from a bibliographic database [3]. To use *prefer* capabilities with *monk*, there are only two additional things you need to know. First, the `|reference` command passes a partial reference to *prefer*, as in: `|reference( Derr Prefer)`. When this command is used, *monk* will automatically invoke *prefer*. Second, the `–R` option to *monk* serves to specify a local or private bibliography file:

```
monk –Sproof –R bibliofile myfile
```

6. Advanced Monk

6.1 Preparing a large document

When working on large documents, it is important to be able to run off parts of the document separately, while maintaining correct numbering and references to other parts of the document. You can do this with *monk* by dividing the document into several files and identifying each file appropriately.

The top-level file should contain all style information and include the various segments. For a book with three chapters, the parent file `myfile` might look like:

```
|make(kluwer)
|style(page_width 7i)
|insert(chapter1)
|insert(chapter2)
|insert(chapter3)
```

The first line of each chapter file should identify it as a part of `myfile`: `|part(myfile)`.

When you run off the entire document, *monk* will remember the assorted section, display, and page numbers. These numbers will then be used when you run off individual parts. Note that the numbers should always be reasonable approximations, but may not always be correct. When several sections are changing simultaneously, you will need to generate the entire document periodically to guarantee that all numbers are exact.

6.2 Modifying the standard formats

You can redefine existing *monk* commands, such as `|ret_address` and `|p`, and you can create new document types using the standard *monk* database language. An early description of the database language is available [10], but there is no up-to-date guide. When the dust has settled, a new manual will be written. Currently, any augmentations you write should be considered temporary.

You can also cheat by adding commands in the underlying formatting language [7], [4]; this method may offer quick fixes, but there is a possibility of conflict with the *monk* program. *Monk* expends considerable effort managing the state of various attributes such as fonts and indentation level. Therefore, if you change things without *monk*'s knowledge by using *troff* directly, *monk* may well undo your changes in the normal course of processing its own input commands.

6.3 Other related programs

There are three other programs useful to *monk* users: *demonk* is the moral equivalent of *deroff*, *monkmerge* is the functional equivalent of *soelim*, and *monkspell* is the *spell* preprocessor. These tools are helpful when processing *monk* files using other programs, such as indexing tools.

7. References

1. Browning, S.A. Cip User's Manual: One Picture is Worth a Thousand Words. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
2. Cherry, L.L. and Lesk, M.E. Tbl — A Program to Format Tables. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
3. Derr, M.A. Formatting References with Prefer. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
4. Kernighan, B.W. A Troff Tutorial. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
5. Kernighan, B.W. Pic — A Graphics Language for Typesetting. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
6. Kernighan, B.W. and Cherry, L.L. Typesetting Mathematics — User's Guide. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
7. Kernighan, B.W. and Ossanna, J.F. Troff User's Manual. In *Unix Programmer's Manual, Tenth Edition*, AT&T Bell Laboratories, 1989.
8. Murrel, S.L. and De Baer, D. `Vtbl` User Manual: bitmaps illustrating how to build tables. AT&T Bell Laboratories internal memorandum, November 16, 1988.
9. Murrel, S.L. and Kowalski, T.J. Overview of Monk 0.2: High-Level Text Compiler. AT&T Bell Laboratories internal memorandum, December 10, 1984.
10. Murrel, S.L. and Kowalski, T.J. Overview of Monk 0.2: Typographical Database. AT&T Bell Laboratories internal memorandum, December 10, 1984.
11. Murrel, S.L. and Kowalski, T.J. Typing Documents on the UNIX System: Using Monk 0.3. AT&T Bell Laboratories internal memorandum, December 10, 1985.
12. Tartanella, C. *UNIX Text Editing and Phototypesetting Starter Package*. Computing Information Service Libraries and Information Systems Center, Bell Laboratories, 1980.

Appendix A

List of Commands

A.1 Global Commands

`A_list` alphabetized list using A.
`I_list` numbered list using I.
`a_list` alphabetized list using a)
`appendix` numbered appendix; can be saved†
`approved` approved
`around` keep a block together here or move it to a later page
`around_placement` outputs all arounds
`authordagnote` footnote marked with a dagger for author names in titlebox
`b` bold
`bargraph` a bar graph in bar
`bargraph_caption` the bar graph's caption; can be saved
`bi` bolditalics
`big` increase point size
`blank_space` leave this much blank space
`bold` alias for b
`bolditalics` alias for bi
`bullet_list` bulleted list using •
`center` center each line
`center_block` center all lines as a block
`chapter` numbered chapter; can be saved
`column_style` width, offset
`comment` comment out text
`computeroutput` figure using cw
`computeroutput_caption` the computer output's caption; can be saved
`constant_width` alias for cw
`cw` constant width
`dagnote` footnote marked with a dagger
`dash_list` dashed list using —
`e` an inline equation in eqn
`endnote` numbered endnote
`endnote_placement` outputs endnotes
`equation` an equation in eqn
`equation_caption` the equation's caption; can be saved
`equation_number` the equation's number; can be saved
`figure` a figure
`figure_caption` the figure's caption; can be saved
`footnote` numbered footnote
`full` use full page width for footnotes and arounds
`graph` a graph in grap
`graph_caption` the graph's caption; can be saved
`here` keep a block together on this page or start a new page
`i` italics
`i_list` numbered list using i)
`ideal` a picture in ideal
`ideal_caption` the ideal's caption; can be saved
`indent` indent by standard amount
`index` save this item and page number for index
`index_level0` index program top level without sublevels
`index_level1` index program top level with sublevels
`index_level2` index program sublevel
`index_placement` put the index here
`insert` insert a file here
`ipa` international phonetic alphabet
`italics` alias for i
`item` item of a list; can be saved
`left_adjust` left adjust the text
`left_center_right` left justify, center, right justify a line
`narrow` use current column for footnotes and arounds
`new_column` skip to the next column
`new_page` skip to the next page
`notation` notation
`number_list` numbered list using 1.
`p` alias for paragraph
`p_a_list` alphabetized list using (a)
`p_number_list` numbered list using (1)
`paragraph` n, i, f, numbered, indented, flushed
`part` multi-part document
`ped` a picture in ped
`ped_caption` the ped's caption; can be saved
`picture` a picture in pic
`picture_caption` the picture's caption; can be saved
`quotation` indent left and right
`r` roman
`reference` a reference in prefer
`reference_database` use this database file for prefer
`reference_include` these reference in prefer
`reference_list` format these databases in prefer
`reference_placement` produce references here
`remember` remember the important thing or page number saved in this label
`roman` alias for r
`s` alias for small
`save` save last important thing, shown as ''can be saved'', in this label
`savepage` save last page number in this label
`savestring` save string under this name

† important values can be saved and remembered; see section 5.18

`section` numbered section; can be saved
`set_counter` page, chapter, section, subsection, subsubsection, appendix, figure, table, footnote, note
`set_string` page_headers, even_headers, odd_headers, page_footers, even_footers, odd_footers
`small` reduce the point size
`source` insert a file here without processing it
`sp` special characters
`starnote` footnote marked with a star
`style` controls general style of document
  `1_5_space` 1 and a half space
  `double_space` double space
  `font_family` times, palatino, bembo, optima, souvenir, helvetica, cw, gothic, condensed, narrow
  `font_size` font_size font_spacing
  `list_spacing` on, off
  `multi_column` multiple columns per page
  `one_column` full page output
  `page_length` length of the printed page
  `page_number` arabic, roman
  `page_offset` offset of the printed page
  `page_width` width of the printed page
  `paragraph_style` indented, numbered, flushed
  `reference` acm, apa, att, ieee, lsa, pami, spectrum
  `reference_option` tm, rp, nroff, troff, sort, sequence, a, d, t, ad, at, da, dt, ta, td, adt, atd, dat, dta, tad, tda
  `single_space` single space
  `triple_space` triple space
  `two_column` two columns per page
`subsection` numbered subsection; can be saved
`subsubsection` numbered subsubsection; can be saved
`table` a table in tbl
`table_caption` the table's caption; can be saved
`tc_entry` tc program entry
`tc_heading` tc program heading
`tc_placement` output the table of contents
`text` fill the text
`u` underline
`underline` alias for u
`unnumbered_appendix` unnumbered appendix
`unnumbered_chapter` unnumbered chapter
`unnumbered_footnote` unnumbered footnote
`unnumbered_section` unnumbered section
`unnumbered_subsection` unnumbered subsection
`unnumbered_subsubsection` unnumbered subsubsection
`verbatim` text exactly as it is

A.2 Letters

`address` who to address the letter to
`body` body of letter
`closing` salutation
`date` date
`greeting` greeting
`ret_address` return address
`signature` signature

A.3 Memoranda

`abstract` abstract for coversheet
`author` name, initials, location, department, extension, room for title box and coversheet
`date` date of tm for title box and coversheet
`document` number, file_case, work_program for title box and coversheet
`keywords` keywords for cover sheet
`signature` signatures and typing credits
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.4 Merit Review Form 1

`date` date for signature
`department` department for titlebox
`name` name for titlebox and signature
`p` paragraph
`signature` signature
`titlebox` title box

A.5 Bi-Annual Research Reports

`author` name, number for footer and title box
`date` report, range for title box
`department` name, number for title box
`p` alias for paragraph
`paragraph` i, indented
`section` section
`signature` signature
`titlebox` title box

A.6 Center Phone Books

`date` date of center phone book
`department` a department
`department_title` title of department
`employee` name, nicname, phone, room, email, secretary, consultant, contractor, intern, limited term, resident visitor, casual, post doc, temporary, mit coop, department
`laboratory` a laboratory

`laboratory_title` title of laboratory
`room` name, phone, room, email
`titlebox` title box

A.7 Technical Memoranda

`abstract` abstract for coversheet
`add_totals` text, other, figure, table, reference counts for cover sheet
`att` is_release for cover sheet
`author` name, initials, location, department, extension, room for title box and coversheet
`cover_sheet` produce the coversheet here
`date` date of tm for title box and coversheet
`dh_name` department head's name for cover sheet
`director_name` director's name for cover sheet
`distribute_complete_memo` distribution list for complete memo
`distribute_cover_sheet` distribution list for cover sheet
`document` number, file_case, work_program for title box and coversheet
`earlier` replacing earlier document_number for cover sheet
`government_security` government security clearance for cover sheet
`keywords` keywords for cover sheet
`mercury` chm, cmm, cmp, elc, lfs, man, mas, mkt, phy, std mercury distribution for cover sheet
`other` end of text body/start of attached pages
`proprietary_class` proprietary classification for cover sheet
`signature` signatures and typing credits
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.8 Internal Memoranda

`abstract` abstract for coversheet
`add_totals` text, other, figure, table, reference counts for cover sheet
`att` is_release for cover sheet
`author` name, initials, location, department, extension, room for title box and coversheet
`cover_sheet` produce the coversheet here
`date` date of tm for title box and coversheet
`dh_name` department head's name for cover sheet
`director_name` director's name for cover sheet
`distribute_complete_memo` distribution list for complete memo
`distribute_cover_sheet` distribution list for cover sheet
`document` number, file_case, work_program for title box and coversheet
`earlier` replacing earlier document_number for cover sheet
`government_security` government security clearance for cover sheet
`keywords` keywords for cover sheet
`mercury` chm, cmm, cmp, elc, lfs, man, mas, mkt, phy, std mercury distribution for cover sheet
`other` end of text body/start of attached pages
`proprietary_class` proprietary classification for cover sheet
`signature` signatures and typing credits
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.9 Technical Correspondence

`abstract` abstract for coversheet
`add_totals` text, other, figure, table, reference counts for cover sheet
`att` is_release for cover sheet
`author` name, initials, location, department, extension, room for title box and coversheet
`cover_sheet` produce the coversheet here
`date` date of tm for title box and coversheet
`dh_name` department head's name for cover sheet
`director_name` director's name for cover sheet
`distribute_complete_memo` distribution list for complete memo
`distribute_cover_sheet` distribution list for cover sheet
`document` number, file_case, work_program for title box and coversheet
`earlier` replacing earlier document_number for cover sheet
`government_security` government security clearance for cover sheet
`keywords` keywords for cover sheet
`mercury` chm, cmm, cmp, elc, lfs, man, mas, mkt, phy, std mercury distribution for cover sheet
`other` end of text body/start of attached pages
`proprietary_class` proprietary classification for cover sheet
`signature` signatures and typing credits
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.10 Release Papers

`abstract` abstract for coversheet
`author` name, initials, location, department, extension, room for title box and coversheet
`date` date of tm for title box and coversheet

`document` number, file_case, work_program for title box and coversheet
`keywords` keywords for cover sheet
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.11 Association for Computing Machinery Proceedings

`abstract` abstract for coversheet
`author` name, initials, location, department, extension, room for title box and coversheet
`date` date of tm for title box and coversheet
`document` number, file_case, work_program for title box and coversheet
`keywords` keywords for cover sheet
`reference_placement` produce references here
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.12 Conference Proceedings Model Sheets

`abstract` abstract for coversheet
`author` name, initials, location, department, extension, room for title box and coversheet
`date` date of tm for title box and coversheet
`document` number, file_case, work_program for title box and coversheet
`keywords` keywords for cover sheet
`reference_placement` produce references here
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.13 Computer Science Press Books

`abstract` abstract for coversheet
`author` name, initials, location, department, extension, room for title box and coversheet
`date` date of tm for title box and coversheet
`document` number, file_case, work_program for title box and coversheet
`keywords` keywords for cover sheet
`title` title of document for title box and coversheet
`titlebox` produce title box here

A.14 Kluwer Academic Publishers Books

`appendix` numbered appendix; IT
`chapter` numbered chapter; IT
`index_placement` put the index here
`reference_placement` produce references here
`section` numbered section; IT
`tc_heading` tc program heading
`unnumbered_appendix` unnumbered appendix
`unnumbered_chapter` unnumbered chapter
`unnumbered_section` unnumbered section

A.15 Nursing Home Song Sheets

`d` directive to the singers
`m` margin text
`refrain` refrain of the song
`smaller` print the song smaller than normal
`style singers` full instructions
`title` title of the song
`verse` verse of the song

Appendix B

List of Special Characters

The special characters that control spacing and carriage movements are described in section 5.6. All other special characters are listed below. For each character, column `Char` contains the final printed character and column `Name` contains the keywords given to *monk* to specify the character. To print the character ☞, type `|sp(rhand)` in your input file.

B.1 Common special characters

Char	Name	Char	Name
☞	rhand	☜	lhand
‘	open quote	’	close quote
—	Em dash	−	minus
-	hyphen	_	underrule
\\		//	
°	degree	•	bullet
†	dagger	‡	double dagger
§	section		
|	|		
.	.		
™	TM		ATT logo
®	registered	©	copyright

B.2 Greek alphabet

Char	Name	Char	Name
α	alpha	A	ALPHA
β	beta	B	BETA

γ	gamma	Γ	GAMMA
δ	delta	Δ	DELTA
ε	epsilon	E	EPSILON
ζ	zeta	Z	ZETA
η	eta	H	ETA
θ	theta	Θ	THETA
ι	iota	I	IOTA
κ	kappa	K	KAPPA
λ	lambda	Λ	LAMBDA
μ	mu	M	MU
ν	nu	N	NU
ξ	xi	Ξ	XI
ο	omicron	O	OMICRON
π	pi	Π	PI
ρ	rho	P	RHO
σ	sigma	Σ	SIGMA
ς	terminal sigma		
τ	tau	T	TAU
υ	upsilon	Y	UPSILON
φ	phi	Φ	PHI
χ	chi	X	CHI
ψ	psi	Ψ	PSI
ω	omega	Ω	OMEGA

B.3 Mathematical symbols

Char	Name	Char	Name
+	+	−	-
±	+-	∗	*
×	times	÷	divide
≃	-wig	~	wig
≠	!=	≡	==
≥	>=	=	<=
∫	integral	∂	partial
∞	inf	∇	del
¬	not	|	or
∝	prop		
⌈	lceil	⌉	rceil
⌊	lfloor	⌋	rfloor

B.4 Mathematical set symbols

Char	Name	Char	Name
∈	member	∪	cup
∅	empty	∩	cap
⊂	subset	⊃	superset
⊆	isubset	⊇	isuperset

B.5 Miscellaneous

Char	Name	Char	Name
fi	fi	fl	fl
ff	ff	ffi	ffi
ffl	ffl		
_	rule	|	box vertical rule
´	acute	`	grave
○	circle	□	square
|	bold vertical	’	footmark
→	->	–	<-
↑	uarrow	↓	darrow
¼	quarter	½	half
¾	3quarter	¢	cent
TEX	TEX	LATEX	LATEX

Appendix C

List Of Examples

This appendix provides the input text for examples that appear in this *Monk* user guide. They are not intended for a novice user, but for those already familiar with these formatting tools. If you want to learn how to use *tbl* or *pic* do not look at these examples. Please read the guides that are listed in the references. These guides, which are readily available, begin with simple examples.

C.1 Table in section 5.14

```
box;
c1 s1 s
c1 | c1 | c
l1 | l1 | n.
Major New York Bridges
=
Bridge          Designer            Length
_
Brooklyn        J. A. Roebling      1595
Manhattan       G. Lindenthal       1470
Williamsburg    L. L. Buck          1600
_
Queensborough   Palmer &            1182
                Hornbostel
_
                                    1380
Triborough      O. H. Ammann        _
                                    383
_
```

Bronx Whitestone	O. H. Ammann	2300
Throgs Neck	O. H. Ammann	1800
_		
George Washington	O. H. Ammann	3500

C.2 Picture in section 5.16

```
A:  ellipse
    ellipse ht .2 wid .3 with .se at 1st ellipse.nw
    ellipse ht .2 wid .3 with .sw at 1st ellipse.ne
    circle rad .05 at 0.5 <A.nw,A.c>
    circle rad .05 at 0.5 <A.ne,A.c>
    arc from 0.25 <A.w,A.e> to 0.75 <A.w,A.e>
```