Programmer’s Manual. The documents here are grouped roughly into the areas of basics, editing, language tools, document preparation, and system maintenance. Further general information may be found in the Bell System Technical Journal special issue on UNIX, July-August, 1978.

Many of the documents cited within this volume as Bell Laboratories internal memoranda or Computing Science Technical Reports (CSTR) are also contained here.

These documents contain occasional localisms, typically references to other operating systems like GCOS and IBM. In all cases, such references may be safely ignored by UNIX users.

General Works
1. 7th Edition UNIX — Summary.
   A concise summary of the facilities available on UNIX.
   The original UNIX paper, reprinted from CACM.

Getting Started
   An introduction to the most basic use of the system.
   An easy way to get started with the editor.
5. Advanced Editing on UNIX. B. W. Kernighan.
   The next step.
   An introduction to the capabilities of the command interpreter, the shell.
   Describes a computer-aided instruction program that walks new users through the basics of files, the editor, and document preparation software.

Document Preparation
   Describes the basic use of the formatting tools. Also describes “−ms”, a standardized package of formatting requests that can be used to lay out most documents (including those in this volume).
   Describes EQN, an easy-to-learn language for doing high-quality mathematical typesetting.
10. TBL — A Program to Format Tables. M. E. Lesk.
    A program to permit easy specification of tabular material for typesetting. Again, easy to learn and use.
    Describes, among other things, the program REFER which fills in bibliographic citations from a data base automatically.
    The basic formatting program.
    An introduction to TROFF for those who really want to know such things.

Programming
    Checks C programs for syntax errors, type violations, portability problems, and a variety of probable errors.
    Indispensable tool for making sure that large programs are properly compiled with minimal effort.
    Describes the programming interface to the operating system and the standard I/O library.

This volume contains documents which supplement the information contained in Volume 1 of The UNIX†

†UNIX is a Trademark of Bell Laboratories.
   How to use the ADB debugger.

Supporting Tools and Languages
   Converts a BNF specification of a language and semantic actions written in C into a compiler for the lan-
   guage.
   Creates a recognizer for a set of regular expressions; each regular expression can be followed by arbitrary
   C code which will be executed when the regular expression is found.
   The first Fortran 77 compiler, and still one of the best.
   Converts a Fortran with C-like control structures and cosmetics into real, ugly Fortran.
   M4 is a macro processor useful as a front end for C, Ratfor, Cobol, and in its own right.
24. SED — A Non-interactive Text Editor. L. E. McMahon.
   A variant of the editor for processing large inputs.
25. AWK — A Pattern Scanning and Processing Language. A. V. Aho, B. W. Kernighan and
   P. J. Weinberger.
   Makes it easy to specify many data transformation and selection operations.
   A super HP calculator, if you don’t need floating point.
   A front end for DC that provides infix notation, control flow, and built-in functions.
   The ultimate dead language.

Implementation, Maintenance, and Miscellaneous
   How to configure and get your system running.
   What do do when you have to change things.
   How the system actually works inside.
   How the I/O system really works.
   How the PDP-11 compiler works inside.
34. A Tour Through the Portable C Compiler. S. C. Johnson.
   How the portable C compiler works inside.
   Describes UUCP, a program for communicating files between UNIX systems.
36. UUCP Implementation Description. D. A. Nowitz.
   How UUCP works, and how to administer it.
   Hints on how to break UNIX, and how to avoid doing so.
   How the bad guys used to be able to break the password algorithm, and why they can’t now, at least not so
   easily.