

Contents

1	General Concepts	1
1.1	History and Evolution of the Language	1
1.2	Essential Toolkit (Compilers)	2
1.3	Basic Programming Workflow	3
	References	5
2	Fortran Basics	7
2.1	Program Layout	7
2.2	Keywords, Identifiers and Code Formatting	8
2.3	Scalar Values and Constants	10
2.3.1	Declarations for Scalars of Numeric Types	11
2.3.2	Representation of Numbers and Limitations of Computer Arithmetic	12
2.3.3	Working with Scalars of Numeric Types	14
2.3.4	The Kind type-parameter	15
2.3.5	Some Numeric Intrinsic Functions	18
2.3.6	Scalars of Non-numeric Types	18
2.4	Input/Output (I/O)	21
2.4.1	List-Directed Formatted I/O to Screen/from Keyboard	22
2.4.2	Customizing Format-Specifications	25
2.4.3	Information Pathways: Customizing I/O Channels	30
2.4.4	The Need for More Advanced I/O Facilities	36
2.5	Program Flow-Control Elements (if, case, Loops, etc.)	37
2.5.1	if Construct	37
2.5.2	case Construct	40
2.5.3	do Construct	42
2.6	Arrays and Array Notation	48
2.6.1	Declaring Arrays	49
2.6.2	Layout of Elements in Memory	50
2.6.3	Selecting Array Elements	51

2.6.4	Writing Data into Arrays	53
2.6.5	I/O for Arrays	56
2.6.6	Array Expressions	58
2.6.7	Using Arrays for Flow-Control	60
2.6.8	Memory Allocation and Dynamic Arrays	64
2.7	More Intrinsic Procedures	67
2.7.1	Acquiring Date and Time Information.	67
2.7.2	Random Number Generators (RNGs)	68
	References	70
3	Elements of Software Engineering	71
3.1	Motivation.	71
3.2	Structured Programming (SP) in Fortran	72
3.2.1	Subprograms and Program Units	73
3.2.2	Procedures in Fortran (function and subroutine)	75
3.2.3	Procedure Interfaces	78
3.2.4	Procedure-Local Data	87
3.2.5	Function or Subroutine?	90
3.2.6	Avoiding Name Clashes for Procedures	92
3.2.7	Modules	93
3.3	Elements of Object-Oriented Programming (OOP)	97
3.3.1	Solution Process with OOP	98
3.3.2	Derived Data Types (DTs).	99
3.3.3	Inheritance (type Extension) and Aggregation	106
3.3.4	Procedure Overloading	109
3.3.5	Polymorphism	112
3.4	Generic Programming (GP)	113
	References	114
4	Applications	117
4.1	Heat Diffusion	117
4.1.1	Formulation in the Dimensionless System	119
4.1.2	Numerical Discretization of the Problem	120
4.1.3	Implementation (Using OOP).	123
4.2	Climate Box Model	128
4.2.1	Numerical Discretization	131
4.2.2	Implementation (OOP/SP Hybrid).	132
4.3	Rayleigh-Bénard (RB) Convection in 2D.	138
4.3.1	Governing Equations	139
4.3.2	Problem Formulation in Dimensionless Form.	141
4.3.3	Numerical Algorithm Using the Lattice Boltzmann Method (LBM)	144

4.3.4	Connecting the Numerical and Dimensionless Systems of Units	150
4.3.5	Numerical Implementation in Fortran (OOP)	152
	References	161
5	More Advanced Techniques	163
5.1	Multiple Source Files and Software Build Systems	163
5.1.1	Object Files, Static and Shared Libraries	164
5.1.2	Introduction to GNU Make (gmake)	170
5.2	Input/Output	182
5.2.1	Namelist I/O	183
5.2.2	I/O with the NETwork Common Data Format (netCDF)	187
5.3	A Taste of Parallelization	205
5.3.1	Parallel Hardware Everywhere	206
5.3.2	Calibrating Expectations for Parallelization	208
5.3.3	Software Technologies for Parallelism.	211
5.3.4	Introduction to Open MultiProcessing (OpenMP)	212
5.3.5	Case Studies for Parallelization	228
5.4	Interoperability with C	235
5.4.1	Crossing the Language Barrier with Procedures Calls.	236
5.4.2	Passing Arguments Across the Language Barrier	239
5.5	Interacting with the Operating System (OS).	242
5.5.1	Reading Command Line Arguments (Fortran 2003)	242
5.5.2	Launching Another Program (Fortran 2008).	244
5.6	Useful Tools for Scaling Software Projects	245
5.6.1	Scripting Languages	245
5.6.2	Software Libraries	246
5.6.3	Visualization	247
5.6.4	Version Control	247
5.6.5	Testing	248
	References	249