FORTRAN IV: From Novice to Expert

Introduction

FORTRAN IV: From Novice to Expert is a comprehensive guide to the FORTRAN IV programming language, designed for programmers of all levels. This book provides a solid foundation in the fundamentals of FORTRAN IV, covering topics such as data types, operators, control structures, arrays, and subprograms. It also delves into more advanced concepts such as recursion, dynamic memory allocation, and file processing. With numerous examples and exercises, this book is an invaluable resource for anyone looking to master FORTRAN IV.

FORTRAN IV was developed in the 1960s and quickly became one of the most popular programming languages for scientific and engineering applications. It is known for its simplicity, efficiency, and portability.

While other languages have gained popularity in recent years, FORTRAN IV remains widely used in many fields, particularly in high-performance computing and numerical analysis.

This book is designed to help readers learn FORTRAN IV in a clear and concise manner. It starts with the basics of the language, such as data types and operators, and gradually introduces more advanced concepts. Each chapter includes numerous examples and exercises to help readers practice and reinforce their understanding of the material.

By the end of this book, readers will have a thorough understanding of FORTRAN IV and be able to use it to develop their own programs. They will also be familiar with the latest developments in the language, such as the addition of new features and the adoption of new standards.

FORTRAN IV: From Novice to Expert is the perfect resource for anyone who wants to learn FORTRAN IV

or improve their skills in the language. With its clear explanations, numerous examples, and exercises, this book is an invaluable tool for programmers of all levels.

Whether you are a student, a professional programmer, or simply someone who is interested in learning a new language, FORTRAN IV: From Novice to Expert is the perfect book for you.

Book Description

FORTRAN IV: From Novice to Expert is the ultimate guide to learning and mastering the FORTRAN IV programming language. This comprehensive book covers all the essential concepts of FORTRAN IV, from basic syntax and data types to advanced topics such as recursion and dynamic memory allocation. With clear explanations, numerous examples, and exercises, this book is perfect for programmers of all levels.

FORTRAN IV is a powerful and versatile language that has been used for decades to develop scientific and engineering applications. It is known for its simplicity, efficiency, and portability. While other languages have gained popularity in recent years, FORTRAN IV remains widely used in many fields, particularly in high-performance computing and numerical analysis.

This book is designed to help readers learn FORTRAN IV in a clear and concise manner. It starts with the

basics of the language, such as data types and operators, and gradually introduces more advanced concepts. Each chapter includes numerous examples and exercises to help readers practice and reinforce their understanding of the material.

By the end of this book, readers will have a thorough understanding of FORTRAN IV and be able to use it to develop their own programs. They will also be familiar with the latest developments in the language, such as the addition of new features and the adoption of new standards.

FORTRAN IV: From Novice to Expert is the perfect resource for anyone who wants to learn FORTRAN IV or improve their skills in the language. With its clear explanations, numerous examples, and exercises, this book is an invaluable tool for programmers of all levels.

Whether you are a student, a professional programmer, or simply someone who is interested in

learning a new language, FORTRAN IV: From Novice to Expert is the perfect book for you.

Chapter 1: FORTRAN IV Fundamentals

Topic 1: A Brief History of FORTRAN IV

FORTRAN IV is a high-level programming language that was developed in the 1960s. It was designed to be used for scientific and engineering applications, and it quickly became one of the most popular programming languages in these fields. FORTRAN IV is known for its simplicity, efficiency, and portability.

The development of FORTRAN IV was led by a team of scientists and engineers at IBM, including John Backus, who is often considered to be the father of FORTRAN. The first version of FORTRAN was released in 1957, and FORTRAN IV was released in 1962. FORTRAN IV was a major improvement over previous versions of FORTRAN, and it quickly became the standard FORTRAN language.

FORTRAN IV was used to develop a wide variety of scientific and engineering applications, including

programs for weather forecasting, fluid dynamics, and nuclear physics. It was also used to develop business applications, such as accounting and payroll systems.

In the 1970s, FORTRAN IV began to be replaced by newer programming languages, such as C and Pascal. However, FORTRAN IV remained popular in scientific and engineering applications, and it is still used today.

FORTRAN IV is a powerful and versatile programming language that is well-suited for scientific and engineering applications. It is simple to learn and use, and it is very efficient. FORTRAN IV is also very portable, which means that it can be used on a wide variety of computers.

The Impact of FORTRAN IV

FORTRAN IV had a major impact on the development of computer programming. It was one of the first high-level programming languages to be widely used, and it helped to establish the basic principles of programming

languages. FORTRAN IV also helped to popularize the use of computers in scientific and engineering applications.

Today, FORTRAN IV is still used in many scientific and engineering applications. It is a powerful and versatile language that is well-suited for these types of applications. FORTRAN IV is also a very portable language, which means that it can be used on a wide variety of computers.

Chapter 1: FORTRAN IV Fundamentals

Topic 2: Basic Syntax and Structure

FORTRAN IV is a structured programming language, meaning that it uses a hierarchical structure to organize code. This makes it easier to read, understand, and maintain. FORTRAN IV programs are made up of a series of statements, which are executed in the order in which they appear.

The basic syntax of a FORTRAN IV statement is as follows:

```
label statement-type argument-list
```

- **label:** An optional label that can be used to identify the statement.
- **statement-type:** The type of statement, such as an assignment statement, an input/output statement, or a control statement.
- **argument-list:** A list of arguments that are passed to the statement.

For example, the following statement assigns the value 5 to the variable x:

```
x = 5
```

The following statement reads a value from the keyboard and stores it in the variable x:

```
READ *, x
```

And the following statement prints the value of the variable x to the console:

```
PRINT *, x
```

FORTRAN IV also has a number of control structures that can be used to control the flow of execution of a program. These include:

- **IF statements:** IF statements are used to make decisions based on the value of a condition.
- **DO loops:** DO loops are used to repeat a block of code a specified number of times.
- **GOTO statements:** GOTO statements are used to transfer control to another part of the program.

These are just a few of the basic syntax and structure elements of FORTRAN IV. By understanding these elements, you will be able to start writing your own FORTRAN IV programs.

Chapter 1: FORTRAN IV Fundamentals

Topic 3: Data Types and Variables

FORTRAN IV offers a variety of data types to represent different types of data, including integers, real numbers, complex numbers, and characters. Each data type has its own set of rules and operations that can be performed on it.

Integer Data Types

Integer data types are used to represent whole numbers, such as 1, 2, and 3. FORTRAN IV provides several integer data types, including INTEGER*2*, *INTEGER*4, and INTEGER*8. The size of an integer data type determines the range of values that can be stored in it. For example, INTEGER*2 can store values from -32,768 to 32,767, while INTEGER*8 can store values from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.

Real Data Types

Real data types are used to represent real numbers, such as 1.23, 4.56, and 7.89. FORTRAN IV provides several real data types, including REAL*4*, *REAL*8, and REAL*16. The size of a real data type determines the precision and range of values that can be stored in it. For example, *REAL*4 can store values with a precision of 7 decimal digits, while REAL*16 can store values with a precision of 19 decimal digits.

Complex Data Types

Complex data types are used to represent complex numbers, which have both a real and an imaginary part. FORTRAN IV provides a single complex data type, COMPLEX, which can store complex numbers with a precision of 7 decimal digits.

Character Data Types

Character data types are used to represent character strings. FORTRAN IV provides a single character data

type, CHARACTER, which can store strings of up to 255 characters.

Variables

Variables are used to store data in a program. Each variable has a name and a data type. The name of a variable must start with a letter and can be up to 31 characters long. The data type of a variable determines the type of data that can be stored in it.

To declare a variable in FORTRAN IV, you use the following syntax:

```
INTEGER :: variable_name
```

For example, the following code declares an integer variable named `age`:

```
INTEGER :: age
```

You can then assign a value to a variable using the assignment operator (`=`). For example, the following code assigns the value 25 to the variable `age`:

```
age = 25
```

Variables can be used in expressions to perform calculations. For example, the following code calculates the area of a circle using the formula `area = πr²`:

```
REAL :: radius = 5.0
REAL :: area = 3.14159 * radius ** 2
```

Variables can also be used to pass data between subprograms. For example, the following code passes the value of the variable `age` to the subroutine `print_age`:

```
SUBROUTINE print_age(age)
INTEGER, INTENT(IN) :: age
PRINT *, "Your age is", age
END SUBROUTINE print_age

INTEGER :: age = 25
CALL print_age(age)
```

Variables are a fundamental part of FORTRAN IV programming. By understanding how to declare and

use variables, you can write programs that perform a wide variety of tasks.

This extract presents the opening three sections of the first chapter.

Discover the complete 10 chapters and 50 sections by purchasing the book, now available in various formats.

Table of Contents

Chapter 10: FORTRAN IV Applications - Topic 1: Scientific Computing - Topic 2: Engineering Applications - Topic 3: Business Applications - Topic 4: Artificial Intelligence - Topic 5: Graphics and Visualization

This extract presents the opening three sections of the first chapter.

Discover the complete 10 chapters and 50 sections by purchasing the book, now available in various formats.