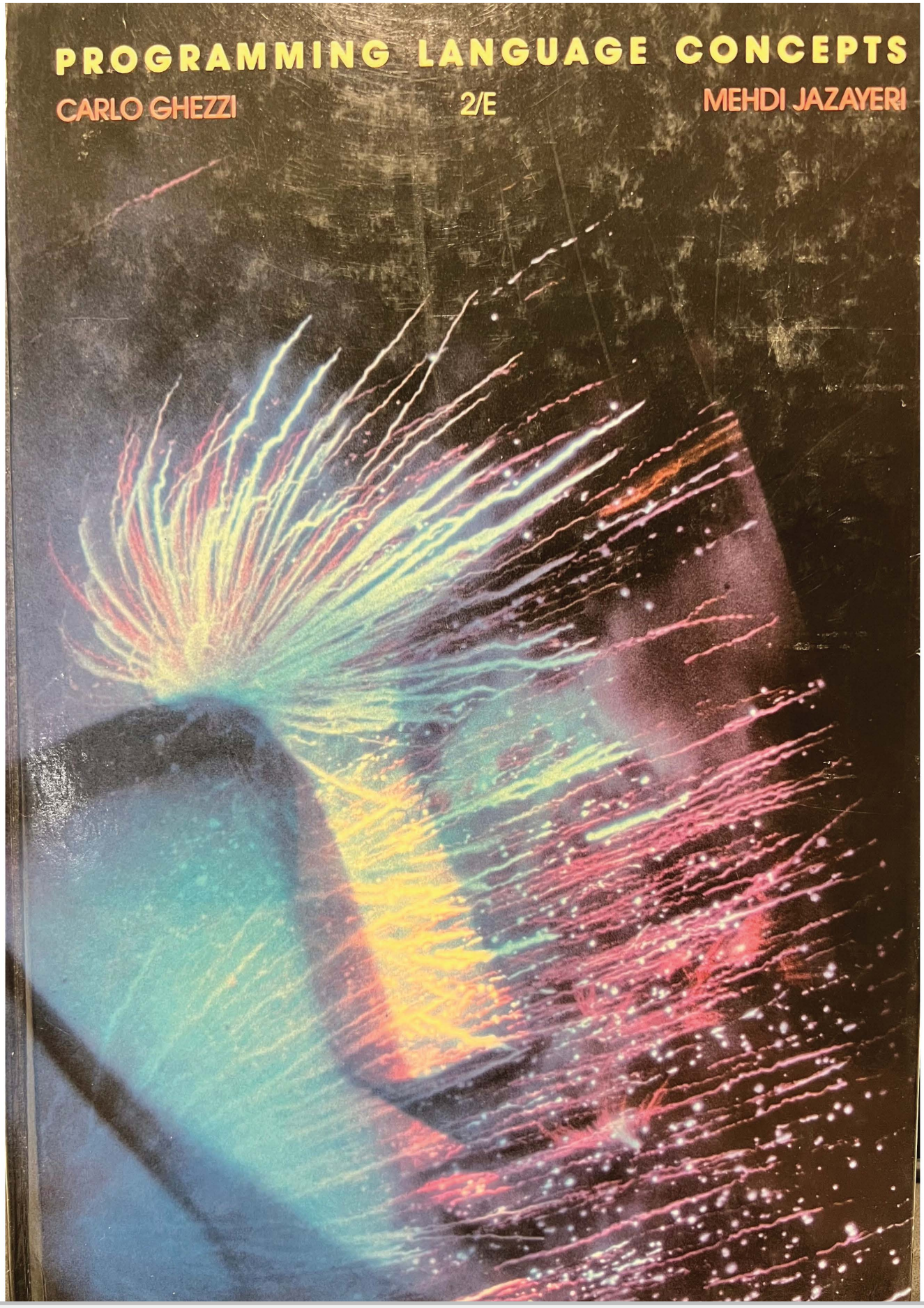
PROGRAMMING LANGUAGE CONCEPTS

CARLO GHEZZI

2/E

MEHDI JAZAYERI



PROGRAMMING LANGUAGE CONCEPTS 2/E

Carlo Ghezzi
Politecnico di Milano

Mehdi Jazayeri
Hewlett-Packard Laboratories

JOHN WILEY & SONS
New York Chichester Brisbane Toronto Singapore

Library of Congress Cataloging in Publication Data:

Ghezzi, Carlo.
Programming language concepts.

Bibliography: p.
Includes index.
1. Programming languages (Electronic computers)
I. Jazayeri, Mehdi. II. Title.
QA76.7.G48 1987 005.13 86-18967
ISBN 0-471-82173-X

Printed in the United States of America

10 9 8 7 6 5 4 3 2

APL. Of compiled languages, ALGOL 68 was the first to allow it in the form of **flex** arrays, followed by CLU. According to the terminology introduced in Chapter 3, such array variables are called dynamic.

4.2.3 Sequencing

A sequence consists of any number of occurrences of data items of a certain component type *CT*. This structuring mechanism has the important property of leaving unspecified the number of occurrences of the component; therefore, it requires the underlying implementation to be able to store objects of arbitrary size (at least in principle).

Strings provide a well-known example of sequences in which the component type is character. The concept of sequence also captures the familiar data-processing idea of a sequential *file*.

It is difficult to abstract a common behavior from the examples of sequences provided by existing programming languages. For example, SNOBOL4 views strings as data objects with a rich set of operations. Conversely, Pascal and C view strings simply as arrays of characters, with no special primitives for string manipulation. Taking somewhat of a middle ground, PL/I and Ada provide string manipulation primitives but, to reduce the problem of dynamic storage allocation, require that the maximum size of a string be specified in the declaration of the string. Files present more serious problems in that they often have peculiar, system-dependent aspects as a result of the necessary interface with the operating system.

Conventional operators on strings include the following.

1. *Concatenation.* The concatenation of *THIS_IS_* and *AN_EXAMPLE* gives *THIS_IS_AN_EXAMPLE*.
2. *Selection of the first (last) component.* Selection of the last component of the above string yields *E*.
3. *Substring.* A substring can be extracted from a given string by specifying the positions of the first and last desired characters.

Simple primitives are usually provided for files. For example, a Pascal file only can be modified by appending a new value to the end of an existing file. Reading is possible only by sequential scanning.

4.2.4 Recursion

A recursive data type *T* can contain components that belong to the same type *T*. To define a recursive type, one can use the type name in the type's definition. For example, type *binary_tree* can be defined as either empty or