Book Review:
Data structures – Structuri de date
by Ion IVAN, Marius POPA and Paul POCATILU (coordinators)

The book entitled *Data Structures* published in Romanian language in two volumes by the ASE Publishing House coordinated by Ion Ivan, Marius Popa and Paul Pocatilu addresses issues particularly important for applied informatics sciences in terms of data structures usage in software development processes.

The team of authors, coordinated by Ion Ivan, Marius Popa and Paul Pocatilu is composed of university professors from the major university centers that include in their specialty Economic Informatics from Romania and Moldova, but also specialists in the field who have extensive expertise in defining, building and using data structures in software development processes from the most important IT & C companies in Romania. Fields of interest of authors include: programming languages and technologies, software engineering, operating systems, data quality and data quality management, software quality and software quality management, information security and project management. Authors' concerns in these areas follow the publication of books, articles in magazines, participation in national conferences and symposia, international development masters courses and coordinating research projects in major research programs conducted at national level.

The activity of the various teams formed within the list of authors of the book *Data Structures*, published in two volumes, follow the publication of books, articles in specialized journals such as Informatica Economică, Economy Informatics, Romanian Journal of Informatics and Automation, Studies and Research of Economic Computation and Economic Cybernetics, Economic Computation and Economic Cybernetics Studies and Research, Romanian Statistical Review and participated with papers at conferences, symposia and workshops, national and international organizations. They also developed and completed research projects in the research programs CNCSIS and CEEX.

During over 1,200 pages, the work *Data Structures* defines the theoretical area in which data structures are defined, built and used, the book presents techniques and methods of programming to implement the defined structures there are given aspects of aggregation processes, management, complexity, security and audit of the data structures problems are proposed and solved and applications are to be solved by readers. These elements are presented gradually over the 37 chapters of the work.

The *Introduction* chapter presents the purpose and need for authors to present this work in the Romanian press, and the trends that are manifested in the approached field.

The next chapter, *Data types and elementary data*, presents aspects of data and information, criteria for the classification of these models, models for the presentation of data, requirements definition, initialization, use of data and the readability of software. The data is represented in calculation by arrays of bits that are initialized by the program or are generated by operations conducted in the program.

The concepts used in C++ are highlighted in the chapter *Object oriented programming*. They are presented and exemplified by the concepts of class objects, instances of classes of objects, data abstraction by classes of objects, arrays of objects, overloading operators, and separation of interface from implementation, implementing the concept of inheritance, the definition of virtual functions, template functions and classes.

In the *Arrays - homogeneous and contiguous data structures* chapter, are highlighted the most important aspects of some of the most frequently used data structures. Operations are also presented and defined by different types of arrays.

A major emphasis is given to the *Functions of processing arrays* whereas matrices calculation involves taking into consideration the details of the programming techniques that lead to anticipated results.

The *Rare matrices* chapter shows the applicability of this type of structure in shaping the business and social processes. There are presented and implemented usual processing operations of rare matrices.

Positioned and ranked data structures, vectors of structures and structures vectors are all presented in the chapter *Article – non-homogenous and contiguous data structure*. It highlights the bal-

ance between increasing the number of dimensions, reducing the fill degree and the complexity of expressions associated with displacement calculations to locate items.

Indirect addressing in C++ is discussed in chapter *Pointer variables*. The chapter shows how to define a pointer variable, pointer arithmetic and the aggregated structures involving pointer variables.

Chapter *Contiguous data joins* shows how to implement the join function in C++. With the help of this data structure the problem of allocating data on the same memory addresses is solved.

In the *Lists – non-contiguous dynamic structures* chapter considerations are presented and data structure operations defined as self-referred are implemented in the C++ language, like lists.

Cases of particular structures of list are highlighted in the chapter *Stacks and queues*. General considerations are presented on these types of data structures and the utility of implemented such structures is exemplified.

Also the category of self-referring structures includes *Binary trees and search trees*. Particular cases in which the use of this type of data structures is needed are presented and exemplified.

Implementation of indexes is done using search trees. In chapter *B trees* are shown specific cases where such structures are implemented. Multi-way search trees are a generalization of binary search trees and the chapter presents and implements the operations specific to this type of structure.

An effort to minimize the data retrieval is performed using balanced tree structures. In the chapter *Balanced trees* there are presented and implemented in the C++ programming language features of balanced tree structures.

The Heap structure and its implementation are outlined in chapter *Heap*. To exemplify the implementation of the structure priority queues and sorting data using the HeapSort algorithm are used.

Another type of data structure presented and implemented in this book is the graph. *Graphs* chapter reviews the main features of these structures and includes implementations in programming language of basic operations on graphs.

Chapter *Dispersion tables* deals with the data structure having the same name. Dispersion tables are used to implement the search algorithms with high efficiency in terms of retrieval time.

External data structures are presented in chapter *Files*. The main categories of files, classified in terms of internal organization and access to components are highlighted.

Accessing items stored by the data structures is done through reference expressions. Defining data structures in a software application is done through defining expressions. Chapter *Data structures defining and reference expressions* consists of developing and usage elements of such expressions.

Chapter *Standard Template Library - STL* includes the main features of standard libraries in the definition and use of data structures. Structure, containers, iterates and algorithms of the STL are highlighted.

The processes of converting data structures are presented in chapter *Conversion of data structures*. Ways of mutual transformation of the most used data structures are highlighted.

Identification of informatics clones is done by computer software which involves the use of data structures. These issues are highlighted and detailed in the chapter *Using the data structures in informatics cloning*.

The relatively limited capacity of the physical devices for storing data requires identifying ways to reduce space occupied by data. Chapter *Compaction and compression of data* shows ways to improve parameters regarding the use of memory units.

Implementation of compaction algorithms and data compression assumes the use of specific data structures. Chapter *Using data structures in data compression* presents algorithms to optimize the memory support used to store data and choosing criteria of the compression algorithm.

The final software product quality depends on the quality of its components, but also how these components are integrated. Qualitative aspects regarding the integration of applications are outlined in Chapter *Aggregation processes of software* which pursues the optimization of software execution.

Software testing is a very important stage in the lifecycle of a software product, being stage which establishes parameters for performance of the developed product. The chapter *Using data structures in software testing* reveals the test process, how these features are unfolded and test particularities of data structures grouped by type.

Management issues are a factor differentiating the quality of software. The chapter *Management of data quality and data structures* addresses issues related to the characteristics of quality, metrics and data cost.

The chapter *Data structures and access in databases* presents issues regarding the storage of data and ways of access to data stored in files with particular organization, files that together with other characteristics define a database.

Mechanisms for retrieval of data are implemented in search algorithms. In chapter *Structures used in the search processes* these mechanisms are highlighted.

Developing applications that implement mechanisms to ensure information security involves the definition and use of "unconventional" data structures. These issues are presented and implemented in the chapter *Data Structures in securing information.*

The chapter *Complexity in data structures* presents models that work with complex data structures. Based on the implemented model, conclusions on efficiency and simplifying models are issued.

Definition of indicators for assessing the area of memory used is achieved in the chapter *Usage degree of the memory area*. The indicators are defined based on the types of data structures.

Structures of type uniform text and associated operations with this type of structure are presented in chapter *Uniform text structures*. The chapter highlights and processes the aggregation and complexity associated with this type of structure.

The chapter *Software inspection* addresses aspects related to information audit, the process of organizing and carrying out audit and inspection achievement automatically using software tools.

XML language is presented in detail and implemented in chapter *The XML data model*. Also, the chapter includes features, classifications and structure of the XML format.

Chapters *Exam tickets given in past years* and *Proposed topics* includes solved applications addressed and proposed to specialist readers, especially for those who prepare in specialties in the field of Economic Informatics.

The *Bibliography* includes a large number of books, articles and papers presented at conferences, national and international symposia, and websites of literature, including contributions from authors in the development field.

The book ends with two annexes which include functions for processing data structures and an analysis of what has been published in the field in international literature.

The book is addressed to specialists in the field of software development, the examples offered by the authors being of real help to those who are specialists in IT & C faculties from the country and the Republic of Moldova.

Prof. Ion Gh. ROŞCA, PhD