

Annex 3B - Topics for the Written Computer Science Exam

1. Algorithms

- 1.1. The notion of an algorithm, characteristics
- 1.2. Data, variables, expressions, operations
- 1.3. Basic structures (sequential, alternative, and repetitive)
- 1.4. Algorithm description (pseudocode programs)

2. Basic elements of a programming language (Pascal or C or C++, by choice)

- 2.1. Language vocabulary
- 2.2. Constants. Identifiers
- 2.3. The notion of data type. Arithmetic, logical, and relational operators
- 2.4. Defining data types
- 2.5. Variables. Declaring variables
- 2.6. Defining constants
- 2.7. Program structure. Comments
- 2.8. Expressions. The assignment statement
- 2.9. Data reading/writing
- 2.10. Control structures (compound statements, alternative and repetitive structures)

3. Predefined subprograms

- 3.1. Subprograms. Parameter passing mechanisms
- 3.2. Predefined procedures and functions

4. Structured data types

- 4.1. Array type
- 4.2. String type - operators, predefined procedures and functions for: reading, displaying, concatenating, searching, extracting, inserting, deleting, and conversions (string ↔ numeric value)
- 4.3. Record type

5. Text files

- 5.1. Text files. Types of access
- 5.2. Procedures and functions for text files

6. Elementary algorithms

- 6.1. Problems operating on the digits of a number in a given numeral system
- 6.2. Divisibility. Prime numbers. Euclid's algorithm
- 6.3. Fibonacci sequence. Calculating sums with a given general term
- 6.4. Finding minimum/maximum values
- 6.5. Sorting methods (bubble sort, insertion, selection, counting sort, merge sort, quicksort)
- 6.6. Merging
- 6.7. Searching methods (sequential, binary)
- 6.8. Algorithm complexity analysis (considering efficiency criteria, execution time, memory space usage, Big O notation)

7. User-defined subprograms

- 7.1. Procedures and functions
 - 7.1.1. Declaration and call
 - 7.1.2. Formal and actual parameters
 - 7.1.3. Parameters passed by value, parameters passed by reference
 - 7.1.4. Global and local variables, scope

7.2. Modular design for problem-solving

8. Recursion

8.1. General presentation

8.2. Recursive procedures and functions

9. Backtracking method (iterative or recursive)

9.1. General presentation

9.2. Generation problems. When it is appropriate to use backtracking

10. Greedy method

11. Divide and Conquer method

12. Generation of combinatorial elements

12.1. Permutations, arrangements, combinations

12.2. Cartesian product, subsets

13. Graphs

13.1. Undirected graphs - terminology, properties, representation methods

13.2. Directed graphs - terminology, properties, representation methods

13.3. Trees - terminology, representation methods in memory

Selected Bibliography

1. M. Frențiu, H.F. Pop, G. Șerban, *Programming Fundamentals*, Ed. Presa Universitară Clujeană, Cluj-Napoca, 2006