Approved:
Software Engineering Master Study Program Committee
2021-02-18

The Entrance Exam Topics to Vilnius University Software Engineering master study programme

I. Programming

1. Data types: simple and structured, build-in types, user-defined types, dynamic structures, abstract data types.
2. Constants, variables, expressions, operators, operations. Control structures: linear, conditional, loops.
3. Memory types, dynamic memory allocation. Pointers.
4. Functions and procedures: purpose, formal and actual parameters, identifiers scope, recursion. Parameter passing methods.
5. Concept of module. Interface and implementation.
6. Exception handling in object-oriented programming languages.
7. Class concept. Encapsulation and information hiding principles.
8. Object concept. Creating, copying, deleting and managing objects. Object lifecycle.
9. Object-oriented programming: composition and inheritance, dynamic binding, method overriding, polymorphism.

Literature:

R. C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, Pearson, 2009
P. Deitel, H.M. Deitel. C: How to Program. 6th ed., Prentice Hall, 2009.
J. Bloch, Effective Java 3rd edition, Addison Wesley, 2017
J. Skeet, C# in Depth, 4th edition, Manning Publications, 2019
G. Booch et al. Object-Oriented Analysis and Design with Applications, 3rd ed., Addison-Wesley Professional, 2007.

II. Database Management Systems

1. Features of Database Management Systems.
2. Relational data model. Data normalization.
3. Semantic data modeling. Entity relationship model and its presentation via relational data model.
4. SQL language for manipulating and querying data in relational databases.
5. Using SQL operations in applications.
6. Managing data safety.
7. Ensuring data consistency.
8. Transactions and their management.

Literature:

C.J. Date. An Introduction to Database Systems. 8th ed., Boston: Addison-Wesley, 2003.
J.D. Ullman, J. Widom. A First Course in Database Systems. 3rd ed. New Jersey: Prentice-Hall, 2007.

III. Algorithms and data structures

1. Concept of algorithm and its formalization.
2. Analysis of algorithm complexity.
3. Searching: linear and binary.
4. Sorting: in-place and out-of-place; algorithms: Selection Sort, Insertion Sort, Bubble Sort, Shellsort, Quicksort, in-place and out-of-place MergeSorts, Radix Sort, Heap Sort).
5. Main data structures (linear list, stack, queue, deck, heap, priority queue).
6. Trees: binary search tree, AVL trees, Red-Black trees, B-tree.
7. Hash tables, collision resolution methods.
8. Graphs.
9. NP-completeness.

Literature:

R. Sedgewick. Algorithms. New York: Addison-Wesley, 1992.
M. Main, W. Savitch. Data Structures and Other Objects, A Second Course in Computer Science (Turbo Pascal Edition), The Benjamin/Cummings Publishing Company, 1995.
T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms, 3rd edition, MIT Press, 2009.
M. Main, W. Savitch. Data Structures and Other Objects Using C++, Addison Wesley The Benjamin/Cummings Publishing Company, 2001.

IV. Software engineering

1. Business analysis based on black and white box principles, applied methods.
2. Software engineering requirements: types and quality attributes.
3. The concept of objective oriented paradigm, UML structure, diagrams and models.
4. Software design, static vs dynamic type systems, system design viewpoints and layers of abstraction.
5. Software development, essential goals, used metrics, testing methods.
6. Software maintenance activities, major challenges and their solving methods, configuration management.
7. Software development project management, essential concepts, roles and groups of processes.
8. Enterprise architecture, Zachman’s framework, architecture management model TOGAF.

Literature:

P. Bourque and R. E. Fairley (Eds.). Guide to the Software Engineering Body of Knowledge (SWEBOK). IEEE Computer Society, 2014.
R.S. Pressman. Software Engineering: A Practitioner's Approach. McGraw-Hill, 2004.
Kim, G., Debois, P., Willis, J., and Humble, J. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. IT Revolution Press. 2016.
Project Management Institute. A Guide to the Project Management Body of Knowledge, 2004.
M. Fowler, K. Scott. UML Distilled. Applying the Standard Object Modeling Language. 3rd ed., Addison-Wesley, 2003.
Object Management Group. Unified Modeling Language Specification, Version 2.5.1. OMG, 2015.

V. Software process

1. The concepts of software process and software process model.
2. The concepts of software process maturity and capability.
3. The staged representation architecture of software process maturity model.
4. The continuous representation of the software process capability model.
5. ISO/IEC 15504 requirements for process dimension.
6. ISO/IEC 15504 definition of process capability dimensions.
7. Software process assessment.
8. Software process improvement.

Literatūra:

Han van Loon. Process Assessment and ISO/IEC 15504. A Reference Book. Springer, 2004.
Han van Loon. Process Assessment and Improvement. A practical guide for managers, quality professionals and assessors. Springer, 2004
M.B. Chrissis, M. Konrad, S. Shrum. CMMI: Guidelines for Process Integration and Product Improvement. 2nd ed., CMMI for Development, v 1.2. SEI series in Software Engineering, Addison-Wesley, 2006.

VI. Human-Computer Interaction

1. User-centered design principles and processes.
2. Usability goals and measures.
3. Context of use analysis.
4. Usability design principles.
5. Information architecture design.
6. User studies.
7. Usability evaluation methods: analytical and empirical.
8. Nielsen's usability heuristics.

Literatūra

David Benyon, Phil Turner, Susan Turner. Designing Interactive Systems: People, Activities, Contexts, Technologies. Addison Wesley, 2014 or 2019.
Helen Sharp, Yvonne Rogers, Jenny Preece. Interaction Design: Beyond Human-Computer Interaction, John Wiley & Sons, 2015 or 2019.
Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs. Designing the User interfaces: strategies for Effective Human-Computer Interaction. Person, 2015 or 2019.