

RTU Course "Algorithmization and Programming of Solutions"

33000 null

General data

| | |
|---|--|
| Code | DIP107 |
| Course title | Algorithmization and Programming of Solutions |
| Course status in the programme | Compulsory/Courses of Limited Choice |
| Responsible instructor | Marina Uhanova |
| Academic staff | Natālija Prokofjeva Oksana Zavjalova Jurijs Ivanovs Olga Jakovļeva Ēvalds Masaļskis Eduards Rēns Marija Dobkeviča |
| Volume of the course: parts and credits points | 2 parts, 9.0 credits |
| Language of instruction | LV, EN |
| Annotation | Students receive academic knowledge in computer applications, as well as practical skills in development of applications using high-level programming languages. The acquired knowledge and skills ensure further successful mastering of computer science courses. |
| Goals and objectives of the course in terms of competences and skills | The goal of the course is to prepare students for further computer science courses, to teach to algorithmize the problem and to check solution by implementing the software. The main objectives are to provide academic knowledge on algorithms, their properties and their development, formal syntax, semantics, data types, structures, data processing operations, as well as to provide practical knowledge about high-level programming languages. The result of the course is archived by developing a set of documented programs of varying complexity. |
| Structure and tasks of independent studies | Individual work consists of: - learning of methodological materials and literature; - development of algorithms; - designing, documenting, coding, and testing of software working individually and in groups. |
| Recommended literature | Obligāta / Obligatory 1. PI katedras datorizētie mācību līdzekļi. RTU ORTUS. 2. Herbert Schildt. Java: beginner's guide. 8th edition New York, McGraw-Hill, 2018. 720 p. Papildu / Additional 1. Herbert Schildt. Java: The Complete Reference. 12th Edition New York, McGraw-Hill, 2021. 1280 p. 2. Rogers Candanhead. Java in 21 days Indianapolis, IN : Sams Pub 2016. - 697 p. 3. Josh Juneau. Java EE7 receipes: a problem - solution approach New York, Apress. 2013. 699 p. 4. Bart Baesens. Beginning Java Programming Indianapolis, IN : Wroox, 2015. 634 p. 5. Robert Sedgewick. Algorithms in Java, Fundamentals, Data Structures, Sorting, Searching. New York, Addison-Wesley, 2002., 768 p. 6. Thomas H. Cormen. Algorithms Unlocked. MIT Press, 2013. 222 p. 7. Marina Uhanova. Programmēšana valodā VBA un VB.NET. Rīga, RTU, 2015. 94 lpp. 8. Programmēšanas e-kursi. www.startit.lv |
| Course prerequisites | Knowledge of informatics course within the scope of secondary school program |

Course contents

| Content | Full- and part-time intramural studies | | Part time extramural studies | |
|--|--|-------------|------------------------------|-------------|
| | Contact Hours | Indep. work | Contact Hours | Indep. work |
| Part 1. Algorithms in daily life and their implementation in computers, development of algorithms, computer program as a representation of algorithm. | 6 | 6 | 0 | 0 |
| Main data types, data processing operations in high-level programming languages. Flow control statements of high-level programming languages. | 6 | 6 | 0 | 0 |
| Organization of loops, one-dimensional and multi-dimensional arrays. | 14 | 14 | 0 | 0 |
| Subprograms their necessity and structure. Subprograms and class libraries. | 12 | 12 | 0 | 0 |
| Algorithms for searching and sorting. The principles of analysis of algorithms. O notation. | 6 | 6 | 0 | 0 |
| Development of the first programs, familiarization with the integrated development environment and debugging tools provided by the development environment. Laboratory work development. | 16 | 16 | 0 | 0 |
| Part 2. Programming paradigms. Introduction to basics of object-oriented programming. | 8 | 8 | 0 | 0 |
| Strings. | 4 | 4 | 0 | 0 |
| Data sets on external storages (files), logical and physical files, serial and direct access. Streams. | 6 | 6 | 0 | 0 |
| Creating and using threads. | 4 | 4 | 0 | 0 |
| Graphical user interface development. Event processing. Graphical functions. | 6 | 6 | 0 | 0 |

| | | | | |
|--|-----|-----|---|---|
| Laboratory work development, participation in group project development. | 32 | 32 | 0 | 0 |
| Total: | 120 | 120 | 0 | 0 |

Learning outcomes and assessment

| Learning outcomes | Assessment methods |
|---|--|
| Part 1. Skills in development of algorithms (including) on the level of flowcharts, skills in software development, skills in debugging. | Successfully completed and defended homework and laboratory works. Successfully completed and defended group projects. |
| Part 2. Skills in complex algorithm development and their implementation using subprograms. | Successfully completed and defended homework and laboratory works. Successfully completed and defended group projects. |
| Knowledge in the basics of analysis and development of algorithms. Is able to analyze and to explain the source code of program, to explain used statements, to locate syntactic and semantic errors. | Successfully passed tests and examination. |

Evaluation criteria of study results

| Criterion | % |
|----------------|-----|
| Lab words | 20 |
| Homeworks | 20 |
| Tests | 10 |
| Group projects | 10 |
| Examination | 40 |
| Total: | 100 |

Study subject structure

| Part | CP | Hours | | | Tests | | |
|------|-----|----------|-----------|------|-------|------|------|
| | | Lectures | Practical | Lab. | Test | Exam | Work |
| 1. | 4.5 | 2.0 | 0.0 | 1.0 | | * | |
| 2. | 4.5 | 1.0 | 0.0 | 2.0 | | * | |