***Discrete Structures***

|  |  |
| --- | --- |
| **COURSE No.:** | **???** |
| **PROGRAM:** | BITL, BBA (elective) |
| **INSTRUCTOR:** | Kalvis Apsītis |
| **CLASS DAYS & TIME:** | **Spring 2020 Semester:** Tuesdays, Thursdays January 6 till April 24, 2020. |
| **OFFICE LOCATION & HOURS:** | near Room 417 by appointment |
| **CONTACT PHONE:** | **+371 29112997** |
| **E-MAIL:** | kalvis.apsitis@rbs.lv |

**TEXTBOOK**

**The Textbook:** Kenneth H.Rosen. Discrete Mathematics and Its Applications, 8th Edition. McGraw-Hill Education. 2019.

**Class Website:** <http://linen-tracer-682.appspot.com/discrete/index.html> (hosting may change – the permanent URL will be announced to the students).   
It contains presentations and descriptions of assignments.   
**Reading Assignments:** Class Website also links to some math papers brief individualized assignments.

**COURSE OBJECTIVES**

At the end of this course, students will be able to prove statements about sets, numbers and discrete structures (everything that can be counted and stored in memory). They will communicate with other humans and computers by defining data structures and their processing rules. This course prepares for advanced algorithms used in computer networks, data encryption, financial transactions, Internet search and social media. The course can strengthen creative thinking and general problem solving capabilities.

**COURSE OVERVIEW**

The course provides foundational material for further studies in computer science. It covers large number of interrelated topics – Boolean and predicate logic, proofs, set theory, recurrence relations, counting methods, graphs and trees.

Since antiquity geometry has been the main tool to teach the methods of formal logic. A classical text “Euclid’s Elements” was studied in gymnasiums and universities. Since 1940-ies logical reasoning is automated with increasingly powerful computing devices. The practical interest about logic has shifted from geometry to other important fields: set theory, predicates and quantifiers, database relations, large integers and their arithmetic to be used in everyday algorithms. This approach is chosen by our course as well.

In order to learn the techniques of Discrete Mathematics we will use these methods and tools:

1. **Class Participation:** The large amount of material requires regular work and diligent practice. It is not easy to “catch up” after missing considerable amount of material. Attending class sessions is mandatory, during that time you should also take part in polls, discussions and other class activities.
2. **Reading and presenting:** Discrete structures are often discussed outside any textbooks; the results can be found in technical papers published by universities. In this assignment you will read a brief result or algorithm. Typically, it covers some topic from our course from another perspective. Your slide material and a 10-minute presentation in front of the class will be evaluated.
3. **Automated tests (About once per week):** The knowledge of discrete math is based on key concepts and techniques. It is not possible to test every kind of mathematical knowledge with multiple-choice tests. Automated tests only check the undersanding of key concepts and routine tasks.
4. **Solving mathematical problems (4 Homeworks):** The key ability developed in this class is the ability to express formal reasoning in English. In this class we suggest that you typeset your first homework in LaTeX using APA style guidelines.
5. **Solving mathematical problems (Midterm and Final Exam):** Essay-type exams provide a useful feedback about your strengths as a problem solver. Computer-based tests and written exams may be preceded by discussions in small teams where students can remind each other some math results (in particular, on topics that will be tested shortly). These discussions are not graded separately, but they can encourage discussion, speaking and listening skills. The participants will be assigned roles in advance – so the students will know, which material they may be asked to cover for their colleagues.
6. **Computations and Proof Checking:** Some topics of discrete mathematics are too time-consuming to be done by hand. Computations (using Python, Scala or some other basic computation environments) may cover manipulation of large integers, experiments in cryptography, counting and graphs. Proof checking on computers (Coq) is another category of computer-based activity – it shows how to formalize your thinking even further, since human-readable proofs often contain omissions.

**COURSE REQUIREMENTS**

In order to get a satisfactory grade, there are some mandatory activities that you need to complete.

1. **Class Participation:** Students will be graded on their attendance and contribution to the class. You should aim at attending all scheduled classes (unless the class is explicitly meant as a remediation class and covers activities that you have already completed). Anyone who misses two classes needs to arrange an individual meeting with the instructor to assess the knowledge gaps and possible remediation steps. One such meeting should be scheduled for every 2 missed classes, even if the missed classes are not adjacent.
2. **Reading and presenting assignment** should be submitted electronically in two parts:   
   (A) Objectives and the outline of your presentation to discuss and receive feedback on these before proceeding with your preparation.  
   (B) Full slidedeck and its delivery in the class.
3. **Automated tests** can be assigned either in class sessions or at other times. You need to complete them from your computer or smartphone by the deadline. You can use notes and computational devices, but cannot contact live humans during the tests.
4. **4 homeworks** should be submitted electronically (typically, on the day preceding the class, where they are discussed). class on the due date and should be submitted electronically. These will include several sets of math problems - mostly proofs or applying your theory knowledge in specific contexts.  
   Late homework submissions are not allowed – you do not get any credit for that.
5. **Midterm and Final Exams.** Books and notes are not allowed during the exams. Make-up exams for the mid-term and the final are generally not given. If there are extenuating circumstances and you must miss an exam the instructor must be notified ahead of the exam time. The only time a make-up exam can be taken is during the week following the date of the exam. If a student does not notify the instructor of an absence or misses the make-up exam deadline, the exam will not be included in the final grade.
6. **Computations** should be completed by the due date and submitted electronically. To get any credit they should pass all the public tests.

**GRADING**

Grading for the course is as follows:

Final grades are calculated on the following basis.

> 949 = 10

850-949 = 9

750-849 = 8

650-749 = 7

550-649 = 6

450-549 = 5

350-449 = 4

< 350 = failing

Your grades will be updated regularly on the online assistant. Please check to see that your grades are recorded correctly.

***Assignments*  Points**

|  |  |
| --- | --- |
| Class Participation | 100 |
| Reading and Presentation Assignment | 100 |
| Automated Tests | 100 |
| Computations and Proof Checking | 200 |
| 4 Homeworks | 200 |
| 2 Written Exams (Midterm and Final) | 300 |
| TOTAL | 1000 |

**ACADEMIC INTEGRITY**

To strengthen ethics within Riga Business School and the business community the RBS policy is to take steps to avoid cases of academic fraud. Be aware that any student who turns in written work that is not original with incorrectly referenced sources (i.e. plagiarized) will be subject to the RBS sanctions policy on Academic Fraud (see the policy on the online assistant). The same consequences apply to academic dishonesty on tests and quizzes.

To find information on what plagiarism is and how to avoid it please visit the links at:  
<http://www.uottawa.ca/plagiarism.pdf>  
<http://www.socialsciences.uottawa.ca/pdf/plagiarism2.pdf>  
<http://www.sass.uottawa.ca/writing/plagiarism.pdf>

This short presentation on research and plagiarism will also help learn to correctly reference sources and provides good advice on research:  
<http://library.acadiau.ca/tutorials/plagiarism/>

During the Discrete Structures class, you should be aware of the following integrity guidelines in particular:

* **Automated tests:** They may be either supervised or unsupervised. You are on your honor to complete them independently (using your notes, Internet and the computational devices, but no human assistance). Failure to do so may result in poor performance during the midterm and final exams.
* C**omputations and Proof Checking:** You should understand all the items you are submitting; you should be capable to explain the code you have written. You are encouraged to discuss your assignments, but it is strictly forbidden to copy chunks of code done by other students or exchange your solutions in any form. Please note that computer code is treated in the same way as essays or any other written work.
* **Homeworks:** Same as with computation exercises – discussing your assignments is encouraged as long as you do not take any notes or copies of the work of other people. At the moment when you write your proofs, you should not interact with others and use only your notes and any other notes and papers. If you happen to find anything useful in the Internet, you need to insert citations.
* **Exams:** Both exams will be proctored and are closed book and closed notes. You can use only paper and pen.

**CLASS SCHEDULE**

A specific schedule of topics is listed below. More specific readings and assignments will be defined before each class.

| Class | Topic | Pre-class preparation | Deliverables and Activities |
| --- | --- | --- | --- |
| Week1 | Propositional logic, proofs, predicates and quantifiers. | Chapter 1 from the textbook. | Doing typesetting with LaTeX and some APA style. |
| Week 2 | Theory of sets, functions, relations, comparing set sizes. | Chapter 2 from the textbook. | Homework 1 is due |
| Week 3 | The theory of algorithms, unsolvable problems, complexity of algorithms | Chapter 3 from the textbook. | Computations and proof checking exercises. |
| Week 4 | Number theory, | Chapter 4 from the textbook. | Homework 2 is due |
| Week 5 | Mathematical induction, recursion, invariants. | Chapter 5 from the textbook. | Computations and proof checking exercises. |
| Week 6 | Counting. The rule o fmultiplication. Permutations. | Chapter 6 from the textbook. | Midterm Exam |
| Week 7 | Discrete Probability. Assigning probabilities. Bayes theorem. Descriptive statistics | Chapter 7 from the textbook. | Computations and proof checking exercises. |
| Week 8 | More counting. Recurrences. Generating functions. Inclusion-exclusion. | Chapter 8 from the textbook. | Homework 3 is due |
| Week 9 | Relations. Types of relations. Orderings. | Chapter 9 from the textbook. | Computations and proof checking exercises. |
| Week 10 | Graphs. Representing graphs. Cycles. Shortest paths. | Chapter 10 from the textbook. | Assign Homework 4 |
| Week 11 | Trees. Tree traversal. Spanning trees. | Chapter 11 from the textbook. | Problem solving workshops |
| Week 12 | Reviewing proof techniques and algorithmic a | Parts of Chapters 12 and 13 from the textbook. | Problem solving workshops |
| Week 13 | Reviewing all the topics |  | Final Exam |