|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. Course title: Programming II. | | | | | |
|  | | | | |
| 2. Code: | | 3. Type (lecture, practice etc.): practice | | | |
|  | | | | |
| 4. Contact hours: 4 hoursper week | | 5. Number of credits (ECTS): 5 | | | |
|  | | | | |
| 6. Preliminary conditions (max. 3):  Programming 1 | | | | | |
|  | | | | |
| 7. Announced:fall semester, spring semester, both | | | | | |
|  | | | | |
| 8. Limit for participants: 48 | | | | | |
|  | | | | |
| 10. Responsible teacher (faculty, institute and department):  Gimesi László PhD (Faculty of Science, Institute of Mathematics and Informatics) | | | | | |
|  | | | | |
| 11. Teacher(s) and percentage: | | Makkai Géza PhD | | 100 % | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | | | | |
| 12. Language:English | | | | | |
|  | | | | |
| 13. Course objectives and/or learning outcomes:  Students will know basic programming structures, software development methodology, and more important programming environments. They will be introduced to C++ programming language and the basics of programming.  With the help of acquired methods students lay down the foundation for their further studies in programming. They will be able to practice algorithmic thinking, programming basic algorithms, and designing, coding, testing and managing individual programming tasks. | | | | | |
|  | | | | |
| 14. Course outline   1. Summarizing previous course, Annunciation of course requirements. 2. Structure types, declaring and using them. 3. Examples for structures, structure type variables, parameters. 4. File and periphery management. 5. Serial file management. 6. Binary file management. 7. Data transfer between programs, parametric program launch. 8. Dynamic memory management, list structures. 9. Indexed file management. 10. Basics of object oriented programming. 11. Creating classes, instantiation, inheritance. 12. Examples for objects. 13. Week 13 Summary, evaluation of course fulfillment. | | | | | |
|  | | | | |
| 15. Mid-semester works   1. Test. 2. Test. 3. Test. 4. Assessment: individual project – development of a registration software | | | | | |
|  | | | | |
| 16. Course requirements and grading  The final grade will consist of the average of the followings:   * 2 assessments: 70% * 3 tests: 0%, * Week 13 assessment: 0% (no grade, but compulsory for fulfilling the course)   Replacement or correctional tests are possible on the last week, or the first week of the exam period. | | | | | |
|  | | | | |
| 17. List of readings   1. C++ Language – Tutorials: http://www.cplusplus.com/doc/tutorial/ 2. Published tasks in [Neptun Meet Street](javascript:__doPostBack('upChooser$btnKollab','')" \o "Neptun Meet Street) | | | | | |
|  | | | | |
| 18. Recommended texts, further readings   1. Any Internet publication about C++ | | | | | |
|  | | | | |
| **Date** | 24 April, 2017 | **Prepared by** |  | | |
| Dr. Gimesi László  Responsible teacher | | |
|  | | | | |
| **Endorsed by** | | |  | | |
| László Tóth, PhD  program supervisor | | |