GHENT UNIVERSITY

Course Specifications

Valid in the academic year 2023-2024

Algorithms (E761042)

Course size *(nominal values; actual values may depend on programme)*

Credits 6.0 **Study time 180 h**

Course offerings and teaching methods in academic year 2023-2024

A (semester 2)	Dutch	Gent	lecture	
			practical	0.0h

Lecturers in academic year 2023-2024

Simoens, Pieter	TW05	lecturer-in-charge

Offered in the following programmes in 2023-2024	**crdts**	**offering**
Bachelor of Science in Engineering Technology(main subject Information Engineering Technology)	6	A
Linking Course Master of Science in Information Engineering Technology	6	A
Preparatory Course Master of Science in Information Engineering Technology	6	A

Teaching languages

Dutch

Keywords

Algorithms, Data structures

Position of the course

The performance of many programs doesn't depend only on the speed of the computer, but often even more on the use of efficient algorithms and data structures. This course provides student the knowledge on fundamental algorithms, algorithmic design methods and advanced data structures. It builds upon the course on Data Structures, and prepares the students for the follow-up course on Advanced Algorithms

Contents

An extensive survey of fundamental algorithms and data structures, together with an analysis of their performance:

- Complexity analysis of algorithms
- Recursion: types, implementation, application to binary search
- Most important algorithms for sorting and selection
- Advanced data structures: red-black trees, multidimensional data structures
- Algorithm design principles: divide-and-conquer, greedy algorithms, dynamic programming, backtracking, randomized algorithms
- P vs NP
- Meta-heurstics: simulated annealing, genetic algorithms

During the lectures, theoretical foundations are provided and applied immediately in exercises to be solved on paper. Programming exercises are designed to foster critical thinking and to cover problems where the seen algorithmic patterns must be applied to various problems.

Initial competences

- Good knowledge of data structures, as covered in the course "Data structures"
- Elementary programming skills in C++, as realized in the course "Programming in C/C++".

Final competences

1 To analyse the complexity of an algorithm using the RAM model. Write down and solve recursive functions of execution time.
2 To assess and improve the runtime and memory usage of an implementation.
3 To have knowledge of the most important algorithms for sorting and selection.
4 To be able to design efficient algorithms, using design principles such as divide-and-

conquer, greedy algorithms, randomisation, dynamic programming and backtracking.
5 To be able to apply advanced data structures in algorithmic components.
6 To recognize and reduce NP-complete problems.
7 To apply heuristic search methods.

Conditions for credit contract

Access to this course unit via a credit contract is determined after successful competences assessment

Conditions for exam contract

This course unit cannot be taken via an exam contract

Teaching methods

Lecture, Practical

Extra information on the teaching methods

- **Theory:** Lectures and paper exercises
- L**abs:** Programming exercises on PC.

Learning materials and price

Syllabus en slides. Estimated cost: 15 EUR

References

- SKIENA S., The Algorithm Design Manual, 3rd ed., Springer, 2020.
- SEDGEWICK R., WAYNE K., Algorithms, 4th edition, Addison Wesley, 2011
- CORMEN T.H., LEISERSON C.E., RIVEST R.L., en STEIN C., Introduction to Algorithms, 3rd ed., MIT Press, 2009, Cambridge MA.
- SEDGEWICK R., Algorithms in C++, Parts 1-4: Fundamentals, Data Structures, Sorting, Searching, Part 5: Graph Algorithms, 3rd ed., Addison-Wesley, 1998, Reading, MA.
- WEISS M.A., Data Structures and Algorithm Analysis in C++, 3rd ed., Addison- Wesley, 2006, Reading, MA.
- GOODRICH M., TAMASSIA R., GOLDWASSER M. Data Structures and Algorithms in Python, Wiley, 2013.
- LEVITIN A., Introduction to the Design and Analysis of Algorithms, 3rd ed., Addison- Wesley, 2012, Reading, MA.
- MANBER U., Introduction to Algorithms. A Creative Approach, Addison-Wesley, 1989, Reading, MA.

Course content-related study coaching

Lecturers are available for additional explanations during the labs. Additional feedback can be obtained after making an appointment.

Assessment moments

end-of-term and continuous assessment

Examination methods in case of periodic assessment during the first examination period

Skills test, Written assessment

Examination methods in case of periodic assessment during the second examination period

Skills test, Written assessment

Examination methods in case of permanent assessment

Skills test

Possibilities of retake in case of permanent assessment

examination during the second examination period is possible in modified form

Extra information on the examination methods

- periodic evaluation: written exam covering theory and exercises + programming exercise on PC
- permanent evaluation: on or more programming assignments on PC

Calculation of the examination mark

In the first assesment period:
- written exam (50%)
- programming exercise during the exam period (30%)
- programming exercises during the semester (20%)

In the second assesment period:
- written exam (50%)
- programming exercise during the exam period (50%)

A mark of ten or more out of twenty obtained for the written exam can be transferred between the assessment periods of the same academic year.
A second mark is calculated based on a weighted sum of the scores obtained for the permanent evaluation (programming exercises, weight 20%) and the programming exercise during the assesment period (weight 30%). If this mark is ten or more out of twenty, then this is transferred to the second assessment period. The mark will comprise 50% of the final mark obtained in the second assesment period, replacing the programming exercise of the second assesment period.
When the student obtains less than 8/20 for the written exam, and/ or for the mark calculated from the scores on the programming exercises (permanent evaluation 20% + exam period 30%), they can no longer obtain a pass mark for the course unit as a whole. If the total score does turn out to be a mark of ten or more out of twenty, this is reduced to the highest fail mark (i.e. 9/20)
Students who eschew one or more parts of the assessment can no longer obtain a pass mark for the course unit. Should the final mark be higher than 7/20, it will be reduced to the highest non-passable mark (i.e. 7/20).