

## Discrete Mathematics (E761029)

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

**Credits 3.0** **Study time 90 h**

**Course offerings and teaching methods in academic year 2023-2024**

A (semester 1)	Dutch	Gent	lecture seminar
----------------	-------	------	--------------------

**Lecturers in academic year 2023-2024**

Denert, Marleen	TW05	staff member
van der Hooft, Jeroen	TW05	lecturer-in-charge

**Offered in the following programmes in 2023-2024**

	<b>crdts</b>	<b>offering</b>
<a href="#">Bachelor of Science in Engineering Technology(main subject Electronics and ICT Engineering Technology)</a>	3	A
<a href="#">Bachelor of Science in Engineering Technology(main subject Information Engineering Technology)</a>	3	A
<a href="#">Linking Course Master of Science in Electronics and ICT Engineering Technology(main subject Electronics Engineering)</a>	3	A
<a href="#">Linking Course Master of Science in Electronics and ICT Engineering Technology(main subject Embedded Systems)</a>	3	A
<a href="#">Linking Course Master of Science in Electronics and ICT Engineering Technology(main subject ICT)</a>	3	A
<a href="#">Linking Course Master of Science in Information Engineering Technology</a>	3	A
<a href="#">Preparatory Course Master of Science in Information Engineering Technology</a>	3	A

**Teaching languages**

Dutch

**Keywords**

Logic, sets, groups, modular arithmetic, finite fields

**Position of the course**

Gathering insights and skills in handling fundamental discrete structures and their applications.

**Contents**

- Logic: truth value, truth table, implication, Boolean algebra, predicates, quantification, proof methods (induction, contraposition, contradiction)
- Sets, relations and functions: elementary concepts, cartesian product, injection, bijection, surjection, order relation, equivalence relation
- Group theory: groups, rings, fields, cyclic groups, generator, order, isomorphism, subgroup, quotient group
- Modular arithmetic: greatest common divisor, Euclid's algorithm, Eulers' totient, linear congruences, Fermat's little theorem, Chinese remainder theorem, diophantic equations, private and public encryption
- Finite fields: construction, calculation rules, applications in coding theory

**Initial competences**

Strict prerequisites: none

Advisory initial competences: basic mathematical knowledge and a solid basic Python knowledge

**Final competences**

1 Master, become skillful in and apply elementary concepts from set theory, logic, modular arithmetic and finite fields.

- 2 Analyse and solve exercises and problems from set theory, logic, modular arithmetic and finite fields.
- 3 Independently construct mathematically correct deductive and inductive lines of reasoning.

**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**

Seminar, Lecture

**Extra information on the teaching methods**

Lab session: guided practice of techniques and concepts, and solving of exercises and problems. Both using pen and paper as well as Python-based programming. A solid basis of Python is a must.

**Learning materials and price**

Dutch syllabus, slides, exercises, supplementary material on Ufora  
Required software: a (local) Python distribution or notebook environment, Excel

**References**

Frank Ayres Jr., Theory and Problems of Modern Algebra. Schaum Publishing, 1965.  
Kenneth H. Rosen, Discrete Mathematics and its Applications. McGraw-Hill Education, 2019.  
Normal L. Biggs, Discrete Mathematics. Oxford University Press, 2002.  
Donald E. Knuth, The Art of Computer Programming, volume 2. Addison-Wesley, 1981.

**Course content-related study coaching**

The lecturer is available before and after classroom lectures and practicum.

**Assessment moments**

end-of-term assessment

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

Written assessment

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

Written assessment

**Examination methods in case of permanent assessment****Possibilities of retake in case of permanent assessment**

not applicable

**Extra information on the examination methods**

written examination with pen-and-paper as well as programming exercises on the computer, on campus

**Calculation of the examination mark**

100 % written examination