




5.00 credits

30.0 h + 22.5 h

Q1

| | |
|-----------------------------|---|
| Teacher(s) | Blondel Vincent ;Delvenne Jean-Charles ;Delvenne Jean-Charles (compensates Blondel Vincent) ; |
| Language : | English |
| Place of the course | Louvain-la-Neuve |
| Prerequisites | This course assumes a sufficient mathematical maturity, equivalent to the level of a third year student in engineering. The course is an introduction to algorithmics, treating mainly of non-numerical aspects. A mathematical analysis of the existence and complexity of algorithms for classic problems pertaining to discrete structures and problems. Previous exposition to non-elementary algorithmic questions is useful to the student; other than that, no prerequisite in algorithmics is assumed |
| Main themes | The course is an introduction to algorithms and their complexity from a non-numerical point of view. The principal topic is the mathematical analysis of the existence of algorithms and their complexity on the classical problems of the field. |
| Learning outcomes | <p>At the end of this learning unit, the student is able to :</p> <ul style="list-style-type: none"> • AA1 : 1,2,3 • AA3 : 1,3 • AA4 : 1 • AA5 : 1,2,3,5,6 <p>At the end of this course the student will be able to :</p> <p>1</p> <ul style="list-style-type: none"> • Study exact and approximate algorithms for combinatorial problems from different viewpoints: design, data structure, performance analysis, existence, complexity. • Apply some general techniques (divide and conquer, dynamic programming, etc.) to solve basic algorithmic problems (e.g. sorting) and perform a worst-case or average-case complexity analysis. • Take initiatives to search information relevant for the analysis of a given problem. • Propose original solutions and compare them to available solutions. • Write a report on the proposed and available solutions. |
| Evaluation methods | The students are evaluated through an individual written exam, on the objectives listed above. Moreover the students write homework papers during the term. The grades for the homework amount to 25% of the final grade (in Jan and August). |
| Teaching methods | The course is organised in lessons and homework. No compulsory on-site exercise sessions. |
| Content | a) Illustration on basic algorithms for sorting and the efficient implementation of different data structures of the main concepts of the course, including an analysis of worst case and average case complexity. b) Treatment of important strategies of design of algorithms including divide-and conquer, dynamic programming, greedy methods. c) Probabilistic algorithms. d) Aspects of complexity theory: complexity classes (deterministic, non-deterministic or probabilistic ; uniform or non-uniform) and decidability. e) Quantum computing: Grover's and Shor's algorithms. |
| Inline resources | Moodle page of the course |
| Bibliography | <ul style="list-style-type: none"> • Algorithmics: Theory and Practice, G. Brassard and P. Bratley, Prentice Hall, 1988. • Introduction to Algorithms, T.H. Cormen, C.E. Leieron and R.L. Rivest, MIT Press 1986. • Notes on the Moodle page |
| Faculty or entity in charge | MAP |

| Programmes containing this learning unit (UE) | | | | |
|--|------------------------|---------|--------------|---|
| Program title | Acronym | Credits | Prerequisite | Learning outcomes |
| Master [120] in Mathematics | MATH2M | 5 | |  |
| Master [120] in Electrical Engineering | ELEC2M | 5 | |  |
| Master [120] in Mathematical Engineering | MAP2M | 5 | |  |