Algebraic Combinatorics, Computability and Complexity
Syllabus for the TEMPUS-SEE PhD Course

Dragan Marušič[1] and Stefan Dodunekov[2]

[1] Faculty of Mathematics
Natural Sciences and Information Technologies
University of Primorska
dragan.marusic@upr.si
[2] Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
stedo@math.bas.bg

1 General goals of the course

The main theme of the first module is algebraic combinatorics, with particular emphasis being given to algebraic graph theory. This part of discrete mathematics provides a natural setting for many important applications as well as nice formulations of problems arising not only in other areas of mathematics, but other fields of science (biology, chemistry, computer science or physics, ...) too. We will focus on combinatorial objects admitting certain specific degrees of symmetry, thus allowing fruitful applications of various algebraic methods enhanced with certain combinatorial and topological tools as well. Special emphasis will also be given to the use of software packages such as MAGMA or GAP. These packages are designed to solve computationally hard problems in algebra, combinatorics, geometry and number theory.

Computability theory is central to many areas of theoretical computer science. It originates in the 1930s with the study of the first formal computational models such as Turing machines, Church's-calculus, Post canonical systems and others. The basic properties of the computable functions are established mainly in the works of Kleene. The second module of the course is an introduction to the theory of computability. The considered computational model is based on unlimited register machines. We present the connections between partial computable and partial recursive functions. We consider certain important computable and computably enumerable problems and describe methods for establishing incomputability.

The foundations of the theory of computational complexity are presented in the third module. We discuss properties of the complexity classes

P and NP. We examine certain NP-complete problems and give a proof of Cook's theorem. We consider the class PSPACE and the notion of PSPACE- completeness.

2 Prerequisites

Basic knowledge on Programming languages, Data structures and Algorithms, Mathematical Logic, Graph theory, Discrete mathematics. More specifically: Computability by Deterministic and Nondeterministic Turing machines, Algorithms on graphs, Propositional and Predicate calculus, Arithmetic.

3 Course modules

(20 units each)

3.1 Algebraic Combinatorics

1. Symmetries of combinatorial objects (1 unit)
2. Group actions (3 units)
3. Coherent configurations and association schemes (2 units)
4. Designs and their symmetries (2 units)
5. Automorphism groups of graphs (2 units)
6. Symmetric graphs – graphs satisfying specific symmetry properties (vertex-transitivity, edge-transitivity, arc-transitivity, half-arc-transitivity,) (2 units)
7. Constructions of symmetric graphs (3 units)
8. Structural properties of symmetric graphs (hamiltonicity, semiregularity, (im)primitivity,) (2 ubits)
9. Combinatorial maps and their symmetries (3 units)

3.2 Computability theory

1. Church's Thesis and effective computability (1 unit)
2. Models of computation (3 units)
3. Examples of computable functions (2 units)
4. Primitive recursive functions (2 units)
5. Coding of the pairs and finite sequences (2 units)
6. An numeration of the computable functions, S-m-n theorem (2 units)
7. Universal theorem (3 units)
8. Decidable and semidecidable sets (2 units)
9. Undecidable problems (3 units)

3.3 Complexity theory

1. Time complexity and Space complexity (2 units)
2. Linear speed up (2 units)
3. Deterministic simulation (3 units)
4. The class **P** (1 unit)
5. The class **NP** (2 units)
6. Polynomial time reducibility (1 unit)
7. **NP** completeness (3 units)
8. The Cook-Levin theorem (3units)
9. **NP** complete problems (3 units)

4 Literature

4.1 Algebraic Combinatorics

[1] N.L. Biggs: Algebraic Graph Theory, Cambridge Univ. Press, 1994.

[2] N. L. Biggs, A. T. White: Permutation Groups and Combinatorial Structures, Cambridge University Press, Cambridge, 1979.

[3] W. Bosma, J. Cannon and C. Playoust, The MAGMA Algebra System I: The User Language, J. Symbolic Comput. 24 (1997) 235-265.

[4] P. J. Cameron. Permutation Groups. LMS Student Text 45. Cambridge University Press, Cambridge, 1999.

[5] J. D. Dixon, B. Mortimer, Permutation Groups, Springer-Verlag, New York, 1996.

[6] C.D. Godsil: Algebraic Combinatorics, Chapman & Hall, 1993.

[7] C. Godsil, G. Royle: Algebraic Graph Theory, Springer, New York, 2001.

[8] H. Wielandt, Finite Permutation Groups, Academic Press, New York, 1964.

[9] The GAP Group, GAP - Groups, Algorithms, and Programming, Version 4.4.12; 2008. (http://www.gap-system.org).

Optional Literature:

B. Alspach, J. Liu, On the Hamilton connectivity of generalized Petersen graphs, Discrete Math. 309 (2009), 5461–5473.

M. Conder, P. Dobcsányi, Determination of all regular maps of small genus, J. Combin. Theory Ser. B 81 (2001), 224

E. Dobson, H. Gavlas, J. Morris and D. Witte, Automorphism groups with cyclic commutator subgroup and Hamilton cycles, Discrete Math. 189 (1998), 69-78.

S. Evdokimov, I. Ponomarenko, Permutation group approach to association schemes, Europ. J. Combin. 30 (2009), 1456-1476

M. Giudici, Quasiprimitive groups with no fixed point free elements of prime order, J. London Math. Soc. (2) 67 (2003), 73–84.

L. Lovasz, Combinatorial structures and their applications, (Proc. Calgary Internat. Conf., Calgary, Alberta, 1969), pp. 243-246, Problem 11, Gordon and Breach, New York, 1970.

A. Malnič, Group actions, coverings and lifts of automorphisms, Discrete Math. 182 (1998), 203-218.

M. Muzychuk, I. Kovács, A solution of a problem of A. E. Brouwer, Des. Codes Cryptogr. 34 (2005), 249–264.

C. E. Praeger, Quotients and inclusions of finite quasiprimitive permutation groups, J. Algebra 269 (2003), 329-346.

R. B. Richter, J. Širan, R. Jajcay, T.W. Tucker and M. E. Watkins, Cayley maps, J. Combin. Theory Ser. B 95 (2005), 189-245.

P. Šparl, A classification of tightly attached half-arc-transitive graphs of valency 4, J. Combin. Theory Ser. B 98 (2008), 1076-1108.

4.2 Computability

[1] Cooper, S. Barry, Computability theory, CRC PRESS, 2003

[2] Cutland N., Computability: An introduction to recursive function theory, Cambridge Univerity Press, 1980

[3] Rogers, H, Theory of recursive functions and effective computability, McGraw Hill, 1967

[4] Soare, R. I., Recursively enumerable sets and degrees, Springer, 1987.

4.3 Complexity

[1] Lewis, H. and Papadimitriou, C., Elements of the theory of computation, Prentice Hall, 2nd ed. 1998.

[2] Sipser, M., Introduction to the theory of computation, PWS Publishing company, 1997

[3] Sommerhalder, R., Van Westrhenen S. C., The Theory of Computability: Machines, Effectiveness and Feasibility, Addison Wesley 1987

5 Grading

The basis for grading will be the students' performance in:

- Homework assigned on a regular basis (at least two per course module);
- Final exam. The final grade will be based on the following scheme:
  - Homework - 40 percents
  - Final exam - 60 percents