Name: **Software Design and Development II**		*NEPTUN-code:* *NIXSF2EBNE*	*Number of periods/week:* full-time: 3 lec + 0 sem + 3 lab
Credit: 6 *Requirement:* exam		*Prerequisite:* *NIXSF1EBNE* Software Design and Development I	
Responsible: Sándor SZÉNÁSI, Ph.D.	*Position:* associate professor	*Faculty and Institute name:* John von Neumann Faculty of Informatics Institute of Applied Informatics	

Way of assessment:

- precondition of signature: achievement of tests and project work
- oral exam

Competences

Course descrition:

Introduction to the advanced principles of object oriented programming and commonly used basic data structures. Main competences: Class hierarchy. Inheritance. Constructors and inheritance. Method overriding and hiding. Polymorphism. Abstract classes. Interfaces. Event handling. Delegates. Traditional error handling methods. Exception handling. Generics. Iterators. Simple and sorted linked lists. Linked list variants. Binary search tree. B-tree. Directed and undirected graphs. Spanning tree. Kruskal and Prim algorithm. Graph search algorithms. Depth-first and breadth-first search. Finding the shortest path. Dijkstra algorithm. Finding components. Topological sorting. Hash maps. Backtracking. Dynamic programming. Greedy algorithms. Branch and bound method. Programming paradigms.

Literature

Sándor Szénási: Algorithms, Data Structures II, Óbudai egyetem, 2014 (in Hungarian, electronic notes)
T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein: Introduction to Algorithms (3rd ed.), MIT Press, 2009