

CS 494/594, Experimental Algorithmics

Spring Term, 2013

Wednesdays, 5:05-7:45pm

MK Room 406

Instructor: Mike Langston

Do you ever design algorithms? Do you ever write code? Do you ever measure resources such as time and space? Do ever you plan to publish your work? If you answer “yes” to some of these questions, then this course may be for you. Lectures will be largely discussion format, focusing on general principles, case studies and student projects.

Text: A Guide to Experimental Algorithms, Catherine McGeoch, Cambridge, 2012

From the preface. This guidebook is written for anyone – student, researcher, practitioner – who wants to carry out computational experiments on algorithms and programs that yield correct, general, informative and useful results. Whether the goal is to predict algorithm performance or to build faster and better algorithms, the experiment-driven methodology outlined in these chapters provides insights into performance that cannot be obtained by purely abstract means or by simple runtime measurements. The past few decades have seen considerable developments in this approach to algorithm design and analysis, both in terms of number of participants and in methodological sophistication. In this book I have tried to present a snapshot of the state-of-the-art in this field, at a level suitable for newcomers to computational experiments. The book is aimed at a reader with some undergraduate computer science experience: you should know how to program, and ideally you have had at least one course in data structures and algorithm analysis. Otherwise, no previous experience is assumed regarding the other topics addressed here, which range widely from architectures and operating systems, to probability theory, to techniques of statistics and data analysis. The book takes a nuts-and-bolts approach that would be suitable as a main or supplementary text in a seminar-style course on advanced algorithms, experimental algorithmics, algorithm engineering, or experimental methods in computer science. Several case studies are presented throughout. A companion website makes the files, programs and tools described in the case studies available for downloading. Suggestions for experimental problems and projects appear at the end of each chapter.