

The exam will have five questions, each carrying 20 points. All questions will have parts. The points each part carries should roughly indicate how much time and effort it should need to solve. You are responsible for everything that was covered in class; please see the course website.

Problem 1: This problem will test understanding of graph exploration, algorithms with for and while loops, searching in sorted arrays, etc. You will be expected to analyse code and say what it does, and how long it takes to run.

Problem 2: A problem that uses the divide and conquer strategy. You will be given an incomplete solution, which you will have to complete. You should be able to formally argue the correctness of the solution and bound the running time (e.g., using induction).

Problem 3: You will be given a problem for which you have to design a dynamic programming solution. A very rough outline will be provided for the solution, and you will have to formally define the quantities that you wish to compute, mentioning the base case, and the computation required to obtain each new value. You will be required to state why your algorithm is correct, and how long it takes.

Problem 4: You should be familiar with notions that go into bipartite matching and network flows algorithms, the Ford–Fulkerson algorithm: vertex covers, residual networks, augmenting paths, max–flow, min–cut, etc.

Problem 5: NP–completeness. You are responsible for basic notions such as P, NP, reductions, SAT, Cook–Levin theorem, etc.