Homework 7

CS 4104 (Spring 2014)

Assigned on April 23, 2014.
Submit PDF solutions on Scholar by the
beginning of class on May 5, 2014.

Instructions:

- You can pair up with another student to solve the homework. You are allowed to discuss possible algorithms and bounce ideas with your team-mate. **Do not discuss proofs of correctness or running time in detail with your team-mate.** Please form teams yourselves. Of course, you can ask me for help if you cannot find a team-mate. You may choose to work alone. *Each of you must write down your solution individually, and write down the name of the other member in your team. If you do not have a team-mate, please say so.* **If your solution is largely identical to that of your team-mate or another student, we will return it ungraded.**
- Apart from your team-mate, you are not allowed to consult any sources other than your textbook, the slides on the course web page, your own class notes, the TAs, and the instructor. In particular, do not use a search engine.
- Do not forget to typeset your solutions. *Every mathematical expression must be typeset as a mathematical expression, e.g., the square of n must appear as n^2 and not as "n^2".* Students can use the LaTeX version of the homework problems to start entering their solutions.
- Describe your algorithms as clearly as possible. The style used in the book is fine, as long as your description is not ambiguous. Explain your algorithm in words. A step-wise description is fine. *However, if you submit detailed pseudo-code without an explanation, we will not grade your solutions.*
- Do not make any assumptions not stated in the problem. If you do make any assumptions, state them clearly, and explain why the assumption does not decrease the generality of your solution.
- Do not describe your algorithms only for a specific example you may have worked out.
- You must also provide a clear proof that your solution is correct (or a counter-example, where applicable). Type out all the statements you need to complete your proof. *You must convince us that you can write out the complete proof. You will lose points if you work out some details of the proof in your head but do not type them out in your solution.*
- Describe an analysis of your algorithm and state and prove the running time. You will only get partial credit if your analysis is not tight, i.e., if the bound you prove for your algorithm is not the best upper bound possible.
- For many of the problems in Chapter 7, you will reduce the given problem to one of the flow-related problems we solved in class. Make sure you describe the reduction completely. For example, if you reduce a problem to the maximum network flow problem itself, specify clearly how will convert an input to the original problem into a flow network (nodes, source and sink, edges, directions, and edge capacities). Do not forget to the prove the correctness of your solution in both directions, as we did in class.
- For some of the problems in Chapter 7, you do not have to reduce the problem "all the way" to computing the maximum flow in a network. A reduction to maximum bipartite matching or to minimum cut may suffice. You can then use the polynomial time algorithms we have developed for these problems to solve the original problem.

- For the problems in Chapter 8, please describe the reduction as clearly as you can and make you sure you prove the correctness of the reduction in both directions, as we have discussed in class.

Problem 1 (10 points) Solve exercise 4 in Chapter 7 (page 416) of your textbook.

Problem 2 (10 points) Solve exercise 5 in Chapter 7 (page 416) of your textbook.

Problem 3 (35 points) Solve exercise 6 in Chapter 7 (pages 416–417) of your textbook.

Problem 4 (45 points, 30 points for part(a) and 15 points for part (b))[1] Solve exercise 19 in Chapter 7 (pages 425–426) of your textbook.

Problem 5 (20 points) Solve exercise 1 in Chapter 8 (page 505) of your textbook.

Problem 6 (30 points) Solve exercise 3 in Chapter 8 (pages 505–506) of your textbook.

Problem 7 (50 points = 25 + 25 points) Solve exercise 19 in Chapter 8 (pages 514–515) of your textbook. *Hint:* You can reduce 3-colouring to one problem (I am not saying which) and the other problem to network flow.

Problem 8 (**Extra credit**: 30 points) Solve exercise 6 in Chapter 8 (page 507) of your textbook. *Hint:* think of monotone satisfiability as a covering problem: you have to find a small number of Boolean variables that together satisfy or "cover" all the clauses.

[1] For part(b), you need to modify the solution for part (a), so the 15 points are for this modification.