

## Thoughts

The problems suggested are enclosed to give you some practice with good exam-level questions. Exam-level questions are questions that you should need to do some thinking about, but not a couple of hours worth. The first things that comes to mind should be pretty close to the intended solution.

Besides these suggested problems, the first 10 or so problems of each chapter are generally good practice problems. Two numbers concatenated with an & are related to each other. Do both, one or neither depending on how clear what is going on is to you.

Don't take the number of decidability problems suggested to be indicative of the degree of emphasis on the exam. We just tried to included a few extra problems because there are no sample decidability problems in the book.

## Greedy

K&T CH 4: 1,2,3,6

## Divide and Conquer

K&T CH5: 2,3

## Dynamic Programming

K&T CH6: 2,3,9,13

## Network Flow

K&T CH7: 6,9,14,15\*

\*For 15 (a) you may want to write out a full solution (reduction, runtime, proof) explaining any relevant details from the bipartite matching problem.

## NP-Completeness

K&T CH8: 4\*,6,10(a),18

\*For question 4, an exam question consisting of all four parts of the problem would be considered too lengthy. Realistically, to deal with time constraints, a final exam question would ask you to solve at most two of the four parts of this question.

## Decidability

Prove whether each of these languages is decidable, undecidable, or not recursively enumerable. It may be good to go through an iteration of trying to think what each should be without proof, checking your intuition with us, then flushing out full proofs:

- 1 The language of Turing Machines which output "yes" on some input.
- 2 The language of pairs of Turing Machines that produce the same output on all inputs.
- 3 The language of Turing Machines,  $M$ , such that  $L(M)$  contains at least 4820 elements.
- 4 The language of Turing Machines,  $M$ , such that  $L(M)$  contains at most 4820 elements.

For 5,6,7 you may want to use Rice's Theorem:

**5** The language of Turing Machines,  $M$ , such that  $L(M)$  is an infinite language.

**6** The language of Turing Machines,  $M$ , such that  $1011 \in L(M)$ .

**7** The language of Turing Machines  $M$ , such that  $L(M) = \Sigma^*$ .

### **Approximation**

K&T CH11: 6,9,10

### **Randomized**

K&T CH13: 1,8