

### Homework problems:

**8.1 (due Nov 2, 2006)** We have an  $n \times n$  chessboard. Some of the squares are covered by chess pieces. We would like to know if we can tile the rest of the board using  $1 \times 2$  dominoes. The input is:  $n$ , the size of the chessboard;  $k$ , the number of squares covered by chess pieces; and  $(a_1, b_1), \dots, (a_k, b_k)$ , the positions covered by chess pieces. Give an efficient algorithm which decides whether the rest of the board can be tiled.

(For example if the input is  $n = 100, k = 2, a_1 = b_1 = 1, a_2 = b_2 = 100$  the output is NO.)

**8.2 (due Nov 2, 2006)** A palindrome is a word which reads the same backwards as forwards; for example "civic" is a palindrome. A sequence of numbers  $a_1, \dots, a_t$  is a palindrome if  $a_i = a_{t+1-i}$  for  $i \in \{1, \dots, t\}$ . We are given a sequence  $S = b_1, \dots, b_n$  and would like to find the longest subsequence of  $S$  which is a palindrome. Give a  $O(n^2)$  algorithm for this problem.

(For example, if the input is 1, 9, 2, 9, 3, 2, 1 then the solution is 1, 2, 3, 2, 1.)

**8.3 (due Nov 2, 2006)** Solve the bonus problem 7.3 from the previous homework. (HINT: use the algorithm for maximum weight matching)

---

### Bonus problems:

**8.4 (due Nov 2, 2006)** The *max-weight* of a spanning tree  $T$  is the maximum weight of an edge of  $T$ . A *min-max-weight spanning tree* is a spanning tree with the minimum max-weight. Give  $O(E + V)$  algorithm which finds min-max-weight spanning tree of a given input graph  $G$ .