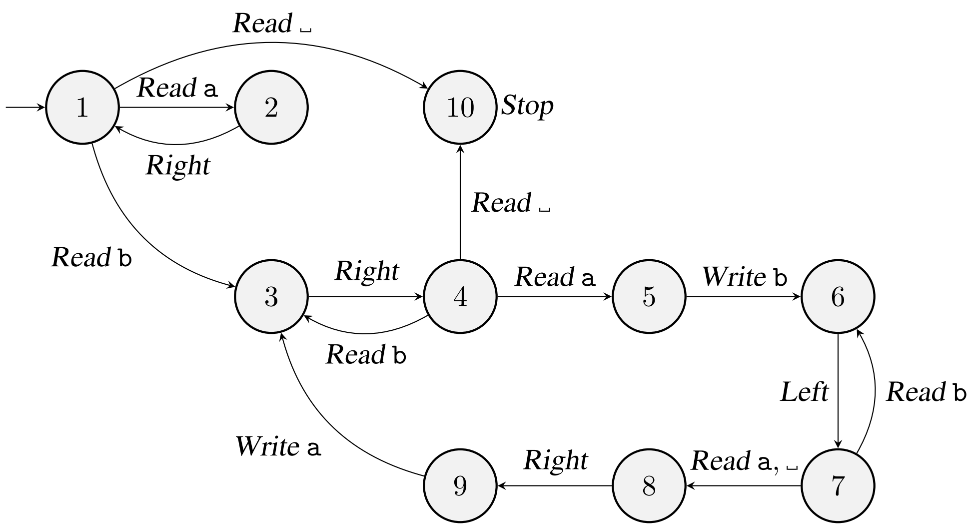
Theories of Computation: Assignment 3

due on Wed 26 March, 12:00

Suzie is a colleague of yours, studying the first year of their Computer Science degree at Birmingham. Because of this, Suzie has been learning about algorithms in their second semester.

Suzie has implemented a simple sorting algorithm on the input alphabet $\Sigma = \{a, b\}$ as the following Turing Machine:



On input of a word $w \in \Sigma^*$ on an otherwise blank tape with the head above the leftmost character, the machine sorts the word and terminates with the head on the space to the right of the word.

Exercise 1. *Trace the execution of the above machine on input of the block* $w =$ `abab`*. At each step, you should state the current tape contents, the position of the machine head, the current state and the next operation to be performed.* ***[5 points]***

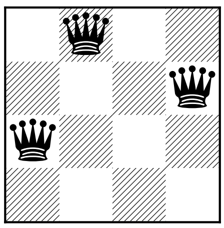
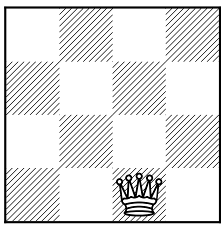
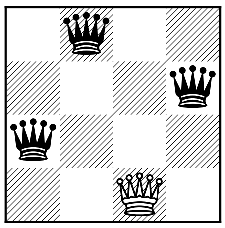
Exercise 2. *Given an input of length $n \in \mathbb{N}$, show that the number of steps (in terms of n) required for the machine to terminate is $O(n^2)$ by giving a precise upper bound on the number of steps in the worst case.* ***[5 points]***

Suzie learned in *Data Structures and Algorithms* that there are algorithms such as bubble sort that can sort a given list in polytime in the length of the given list. However, there are many problems that are not known to have a polytime solving algorithm. One such example is the *n-Queens completion problem*, which Suzie is especially interested in as a lover of the classic board game Chess.

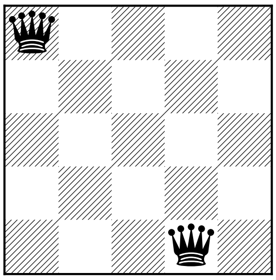
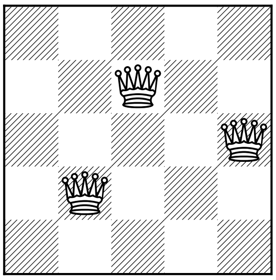
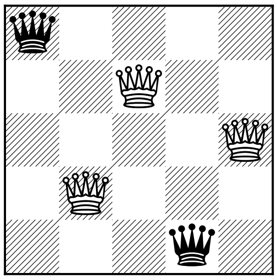
The n-Queens completion *existence* problem is explained as follows: for some natural number $n > 3$, given an n-by-n chessboard (i.e. a square grid made up of $n \times n$ cells) on which $0 \leq m \leq n$ cells are occupied by Queens, return 'Yes' if there is a placement of a further $n - m$ Queens on the board such that, on the resulting chessboard, no two Queens are on the same row, column or diagonal; otherwise, return 'No'.

For example:

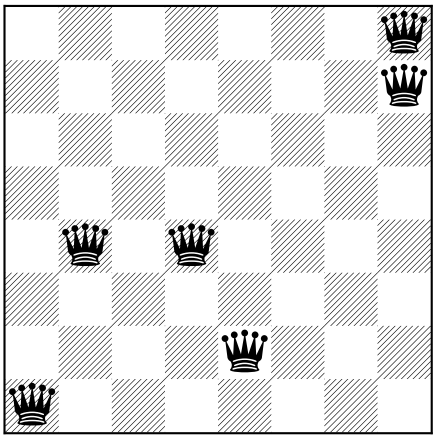
1. Given $n = 4$ and an initial configuration of $m = 3$ Queens as shown in the left image, the answer is 'Yes' because there is a valid placement of the remaining $n - m = 4 - 3 = 1$ Queen (i.e. the one shown in the center image, which creates the resulting chessboard shown in the right image).

2. Given $n = 5$ and an initial configuration of $m = 2$ Queens as shown in the left image, the answer is 'Yes' because there is a valid placement of the remaining $n - m = 5 - 2 = 3$ Queens (i.e. the one shown in the center image, which creates the resulting chessboard shown in the right image).

3. Given $n = 8$ and an initial configuration of $m = 6$ Queens as shown in the left image, the answer is 'No' because there is no valid placement of the remaining $n - m = 8 - 6 = 2$ Queens.



We can encode the chessboards used in the n-Queens completion existence problem using the alphabet $\Sigma = \{\sqcup, Q, \|\}$, where $\sqcup$ represents an empty cell, Q represents a cell occupied by a Queen, and $\|$ is used to delineate the start or end of a row.

For example, the input 8-by-8 chessboard given in the third example can be encoded as the following word:

$$\|\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup Q\|\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup Q\|\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\|\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\|\sqcup Q\sqcup Q\sqcup\sqcup\sqcup\sqcup\|\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\|\sqcup\sqcup\sqcup\sqcup Q\sqcup\sqcup\sqcup\|Q\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\sqcup\|$$

Exercise 3. *Show that the n-Queens completion existence problem is in* ***NP****.*
You should:

1. *Say what a certificate is for this problem (using the encoding above),*
2. *Explain why it has length polynomial in the length of the input encoding word,*
3. *Explain informally how to check that a candidate certificate is indeed a certificate, and explain why the worst case number of steps needed to perform this check is polynomial in the length of the input encoding word.* ***Your argument can be quite informal here – in particular, you are not expected to give a full Turing machine.***

[5 points]

When you submit, look for the word 'Submitted!' – otherwise, you may have not submitted properly.