# Advanced Algorithms: Homework 4

Due on Feb. 14, 2024 at 11:59pm EST

*Professor Dana Randall Spring 2024*

**As stated in the syllabus, unauthorized use of previous semester course materials is strictly prohibited in this course.**

**Note: This should have already been inferred, but unless otherwise stated, justification for a result / design should always be given.**

## Exercise 1

For some problem $P$, two online algorithms $A_1$ and $A_2$ are given, with a competitive ratio of 2 and 3, respectively. Design a randomized online algorithm with competitive ratio 9/4.

## Exercise 2

We consider a version of the game Memory with just one player. As in the usual game, let $n$ pairs of cards lie on the table, face down. In each move of the game, the player can turn two cards, one at a time. If they are identical, they are removed, otherwise they are turned face down again and put back in their spots. The goal of the player is to remove all cards with the least number of moves. Even after the successful turning of a pair, any further turning is considered a move. We assume that the player can remember all cards that have been turned at any time. The optimal strategy needs $n$ moves to remove all $2n$ cards.

(a) Design an online strategy that takes at most $2n$ moves. i.e. that is 2-competitive

(b) Design an online strategy that takes at most $2n - 1$ moves

(c) Show that any deterministic strategy can be forced to take at least $2n - 1$ moves.

(d) What changes about the competitive ratio if the player gets a free move after removing a pair?

## Exercise 3

For the Paging Problem, let ALG be any marking algorithm as presented in the lecture with a cache of size k, and let OPT be an optimal off-line-algorithm with a cache of size $h \leq k$. Prove that ALG is $\frac{k}{k-h+1}$-competitive.

*Hint:* Using the strategy from class where $h = k$, consider a decomposition of the request sequence $\sigma$ into phases that are maximal subsequences with $k$ distinct pages requested.