CS218 HW 1 Challenge

due Thursday, April 11, 23:59 PM

Problem A:

Dynamic Programming approach. Two lists of size $M + 1$: "dp" for maximum points up to limit, and "chosen" for a list of chosen sectors. I only need them to be 1-D because every time I increment num. of darts used, dp[M] is calculated first using dp data from the previous dart amount, saving space.

Loop runs for each dart thrown (4 times), M times for the point limit, and through all n sectors. If including the current sector in the sum would increase the point total, then include it. Output is chosen[M] representing the sectors chosen for the max score given 4 dart throws and a point limit of M.

Runtime: $O(N * M)$ due to loop filling the dp and chosen lists; Space Complexity: $O(M)$, for dp and chosen arrays.

* Note: I tried lots of ways to save on time and space, but couldn't manage on figuring it out. At least, I'm certain that this solution is absolutely correct if given M and N of smaller size.