Exercise 6.5

Saturday, 30 September 2023 18:03

i. An array with displacement of all elements at most $M - 2B$ can be sorted in $O\left(\frac{n}{B}\right)$ I/Os by performing insertion sort. This works in $O\left(\frac{n}{B}\right)$ I/Os, because, in every iteration of the loop, the element which is going to be inserted in the already-sorted sequence will never have to go back further than $M - 2B$ positions. This implies that (even with blocks sticking out), it holds that, for each element which is going to be inserted in the already-sorted sequence, no I/Os need to be done *(except possibly a single I/O to read the block in which the new element is located, if it is located in a block which has not been accessed before)*. In other words, each block in the array needs to be loaded exactly once. This is possible in $O\left(\frac{n}{B}\right)$ I/Os.
ii. The proof for the permutation bound no longer works because the number of valid permutations for the n elements in the array is no longer $n!$, but a much smaller number (given that each element can only be assigned to $2(M - 2B) + 1$ positions, instead of n positions). Thus, the number of valid output states is much lower than $\frac{n!}{(B!)^{\frac{n}{B}}}$.