Lab 7 (19th Jan 2022)

Problem 2*

Several possible solutions. We describe one solution for a slightly more general scenario.

Let S be a set of elements. We want to print all permutations of the elements of S.
The idea is straightforward:

- For each $i \in S$, print i followed by each permutation of the elements in $S \setminus \{i\}$.

The implementation of this simple idea is non-trivial. Here is one possibility:

printPermutations$(\text{set } S, \text{string prefix})$

- If $|S| = 0$ then print prefix and return
- Else
    - FOR each $i \in S$ that is not crossed out, do this once:
        - Cross out i from S.
        - Concatenate i to the end of prefix.
        - Call printPermutations(S, prefix)
        - Uncross out i in S.
        - Remove i from the end of prefix.

The above function can be called with $S = \{1, 2, \ldots, n\}$ and prefix = "" to get the result.

Problem 3

- Pseudocode is trivial. This is an exercise in implementing "call by reference" using pointers.

Problem 4

- Pseudocode is trivial. This is an exercise in declaring a pointer to a pointer.