Basic explanation:https://mvikas886.medium.com/infix-to-postfix-expression-conversion-596f2dbcebdd

https://www.geeksforgeeks.org/arithmetic-expression-evalution/#:~:text=The%20stack%20organization%20is%20very,i.e.%2C%20A%20%2B%20B).

ques

2nd rule: if lower priority and higher come no issue but opposite need to pop out the higher priority one

This is priority for the infix conversion.

See the division operator has higher priority than + hence can stay together.

No 2 operator of same operator can be in the stack. We remove the operator of same priority till new one remains.

See here + and + plus 2 time no issue as not together. And whenever the bracket closes pop the operator in between both.

See here \* is of greater priority than – hence pop it and + and – are same priority and hence cant stay together. Hence pop +.

Now coding logic:

Here to the right is given the logic.

Code:

https://simplesnippets.tech/infix-to-postfix-conversion-using-stack-data-structure-with-c-program-code/

full code.