Problem 1.

Give a regular expression, simplified to the best of your abilities, for the language of **all** strings of a's, b's, and c's where a is never immediately followed by b.

Solution

$$((a^+c^+)^*b^*c^*)^*a^*$$

Reasoning

Given any a, it must be followed by another a, a c or be the end of the string. Thus, the first part of the expression $(a^+c^+)*$ covers the case of a string of a's which is not at the end of the string and thus must be followed by a c.
The case of a string of a's terminating the string is covered at the end of the expression.
Note that with the expression $(a^+c^+)^*a^*$ we have the set of strings, which if they are not empty, start in a and only contain a and c.
We simply follow our optional a's (followed by c's) with as many b's and c's as we want.
Note that they may be in any order, as all of our our groups are zero or more inside another group of zero or more.
Further note that it is perfectly possible for a b to be followed by a a, this is fine and does not violate the terms of the problem.
Thus, although c's are required following occurences of a in the string, we can add individual c's in at any time with c^*

Examples (note groupings are not necessarily unique)
Subscripts denote which group is used to generate each substring $((a^+c^+)_1^*b_2^*c_3^*)^*a_4^*$.

$$\begin{aligned}
a &: a_4 \\
b &: b_2 \\
acacbaaa &: (ac)_1(ac)_1b_2(aaa)_4 \\
cccccbbb &: (ccccc)_3(bbb)_2 \\
acbacb &: (ac)_1b_2(ac)_1b_2 \\
cbacba &: c_3b_2(ac)_1b_2a_4
\end{aligned}$$

Problem 2.

Give a regular expression, simplified to the best of your abilities, for the language of **all** strings of a's, b's, and c's that contain an even number of b's.

Solution

$$((a^*c^*) + (b(a^*c^*)^*b))^*$$

Reasoning

If we read strings from this language left to right, if we find a b then there must be another b separated only be optional a's and c's. That is to say that, we can assume there will be no b's between them, else we would have chosen the closer pair.
Thus the problem can be thought of as a series of $(a^*c^*)^*$, that is, strings of a's and c's, which may or may not be surrounded by b's.

Examples (note groupings are not necessarily unique)
Subscripts denote which group is used to generate each substring $((a^*c^*)_1 + (b(a^*c^*)_3^*b)_2)^*$.

$$\begin{aligned} aaaccc &: (aaa)_1(ccc)_1 \\ baaccaab &: (b(aaccaa)_3b)_2 \\ bbbbbb &: (bb)_2(bb)_2(bb)_2 \\ aacbbbbabbacac &: (aa)_1(c)_1(bb)_2(bb)_2(a)_1(bb)_2(a)_1(c)_1(a)_1(c)_1 \end{aligned}$$

Problem 3.

Simplify (if possible) the expression $(a+b+c)^*(a+b)^*$, then describe as concisely as you can in English the language it defines.

Solution

$$(a+b+c)^*$$

Note that + takes the union of the left and right, in this case just the characters, thus it behaves like an or operator. By this logic, we know that $(a+b) \subset (a+b+c)$. Thus any string in $(a+b)*$ could instead be created by another application of $(a+b+c)*$.

This expression defines a language of string containing the characters $\{a, b, c\}$ (in any order, combination and quantity).

Problem 4.

Simplify (if possible) the expression $(a+b)^*c^*(a+b)^*$, then describe as concisely as you can in English the language it defines.

Solution

Cannot be simplified.

This expression defines the language containing the characters $\{a, b, c\}$ where all characters c must by neighbors of each other, with (any length) combinations of $\{a, b\}$ on either side. Note that the given expression describes exactly that, with no redundancy, and no part can be removed without removing one of those requirements.