The following exercises are designed to test your understanding of recursion. The functions are defined using a variant of LISP known as meta-LISP. In order to aid your understanding, the function defined in problem 1 is identical to the one below:

\[
\begin{align*}
drop(x) &= \text{if null } x \text{ then nil } \\
&\quad \text{else (car } x\text{) cons drop(cdr } x\text{)}
\end{align*}
\]

The idea is that
\[
\begin{align*}
a \cdot x &= \text{car } x \\
d \cdot x &= \text{cdr } x \\
n \cdot x &= \text{null } x \\
at \cdot x &= \text{atom } x \\
a \cdot b &= \text{a cons b} \\
\langle a \rangle &= \text{a cons nil = a list whose single element is } a \\
a \cdot b &= \text{concatenate lists } a \text{ and } b \text{ (i.e. append list } b \text{ to list } a) \\
\text{reverse}[x] &= \text{reverses the top level list } x. \text{ For example reverse}\![\langle A B C \rangle] = \langle C B A \rangle. \text{ But reverse}\![(\langle A B C \rangle \langle D E \rangle)] = (\langle D E \rangle \langle A B C \rangle).
\end{align*}
\]

1. Consider the function \( \text{drop} \) defined by
\[
\text{drop}[x] \leftarrow \text{if n } x \text{ then nil else } [a x]. \text{drop}[d x].
\]
Compute (by hand) \( \text{drop}[\langle A B C \rangle] \). What does \( \text{drop} \) do to lists in general?

2. What does the function \( \text{r2} \) do to lists of lists? How about
\[
\begin{align*}
\text{r2}[x] &\leftarrow \text{if n } x \text{ then nil else reverse}[a x]. \text{r2}[d x] \\
\text{r3}[x] &\leftarrow \text{if at } x \text{ then x else reverse}[\text{r4}[x]] \\
\text{r4}[x] &\leftarrow \text{if n } x \text{ then nil else } \text{r3}[a x]. \text{r4}[d x]?
\end{align*}
\]

3. Compare the following function with the function \( \text{r3} \) of the preceding example:
\[
\begin{align*}
\text{r3'}[x] &\leftarrow \text{if at } x \text{ then x else } \text{r3'}[d x] * \langle \text{r3'}[a x] \rangle
\end{align*}
\]

4. Consider \( \text{r5} \) defined by
\[
\begin{align*}
\text{r5}[x] &\leftarrow \text{if n } x \lor \text{n } d \text{ x then } x \\
&\quad \text{else } [a \text{ r5}[d x]]. \text{r5}[a x]. \text{r5}[d \text{ r5}[d x]]).
\end{align*}
\]
Compute \( \text{r5}[\langle A B C D \rangle] \). What does \( \text{r5} \) do in general. Needless to say, this is not a good way of computing this function even though it involves no auxiliary functions.