CONS, CAR, CDR, LIST

The fundamental data structure in Scheme is called the “cons cell.” It's created with the primitive \texttt{cons}. For example:

\begin{verbatim}
(cons 3 4)
\end{verbatim}

produces a cons cell which we would draw as:

\begin{center}
\begin{tabular}{|c|c|}
\hline
3 & 4 \\
\hline
\end{tabular}
\end{center}

and is in printed form \( (3 . 4) \)

The left-side item is retrieved with the selector \texttt{car}, and the right-side item with \texttt{cdr}; e.g.:

\begin{verbatim}
(define foo (cons 3 4))
(car foo) ; produces 3
(cdr foo) ; produces 4
\end{verbatim}

An aside on the naming of \texttt{car} and \texttt{cdr}:

“Lisp was originally implemented on the IBM 704 computer, in the late 1950s. The 704 hardware had special support for splitting a 36-bit machine word into four parts: an ‘address part’ and ‘decrement part’ of fifteen bits each, and a ‘prefix part’ and ‘tag part’ of three bits each.”

So, \texttt{car} is an abbreviation for “contents of address register,” and \texttt{cdr} is for “contents of decrement register.”

\textbf{Question 1:} What is the printed form of \( (\texttt{cons} \ (\texttt{cons} \ 1 \ 2) \ 3) \)?

\textbf{Question 2:} Draw the data structure \(((\texttt{cons} \ (\texttt{cons} \ 1 \ 2) \ (\texttt{cons} \ 3 \ 4))\). Use arrows to point from the contents of one cons cell to another.
Cons cells are assembled into chains called lists. A list is similar to a singly-linked list data structure in C. A list may be created with the primitive list. For example, \((\text{list } 1 \ 2 \ 3)\) produces:

A list is terminated with what’s called the “empty list.”

- It’s denoted with a slash through the cons cell’s cdr section (per the “box and pointers” diagram above).
- In the book, it’s called \texttt{nil}.
- In Racket, it’s produced with \texttt{'}()\).

**Question 3.** Use a nested expression of \texttt{cons} statements to produce the example list above.

**Question 4.** Assume the list above is bound to the symbol \texttt{foo}. Write an expression that will retrieve the value 2 from \texttt{foo}.