The structure of a binary tree with integers at the leaves is represented by the following ML datatype:

\[
\text{datatype } T = \text{n of int} | \text{d of T*T};
\]

Define a function \( \text{flatten} \) which when given an argument \( t \) of type \( T \) will yield the list of integers obtained by a left to right walk over \( t \). For example,

\[
\text{flatten } (\text{d}(\text{d}(\text{n}1,\text{n}2),\text{n}3)) = [1,2,3].
\]

Define a function \( \text{splits} \) which, when given a list of length \( n > 0 \), will yield the list of 2-tuples representing the \( n - 1 \) ways of splitting the given list into two non-empty sublists. For example,

\[
\text{splits } [1,2,3] = [ ([1], [2,3]), ([1,2], [3]) ].
\]

Hence or otherwise define a function \( \text{alltrees} \) which, when given a list of length \( n > 0 \), will form a list of all the trees of type \( T \) that will flatten to the given list. For example,

\[
\text{alltrees } [1,2,3] = [ \text{d}(\text{n}1, \text{d}(\text{n}2, \text{n}3)), \text{d}(\text{d}(\text{n}1, \text{n}2), \text{n}3) ].
\]