Exercise 1: Fibonacci Heap Worst Cases \textit{(7 Points)}

(a) Consider the Fibonacci heap given below. Perform a \texttt{delete-min} operation. Give the state \textit{before} and \textit{after} the \texttt{consolidate} operation. Conduct the \texttt{link} operations \textit{exactly} in the order in which the algorithm given in the lecture does it (Chapter 5 Part IV Slide 16). \textit{(4 Points)}

(b) Give a valid instance of a Fibonacci heap where \texttt{delete-min} has a \textit{worst case} runtime of $\Omega(n)$ and explain why this is the case for that instance. \textit{(3 Points)}

\begin{center}
\begin{tikzpicture}
\node (1) at (0,0) {1};
\node (2) at (2,0) {2};
\node (3) at (1,-1) {3};
\node (4) at (2,-1) {4};
\node (5) at (1,-2) {5};
\node (6) at (3,-2) {6};
\node (7) at (2,-2) {7};
\node (8) at (1,-3) {8};
\node (9) at (0,-4) {9};
\node (10) at (3,-1) {10};
\node (11) at (1,-1) {11};
\node (12) at (2,-1) {12};
\node (13) at (2,-2) {13};
\node (14) at (1,-2) {14};
\node (15) at (3,-2) {15};
\draw (1) -- (3);
\draw (1) -- (5);
\draw (1) -- (8);
\draw (2) -- (4);
\draw (2) -- (7);
\draw (5) -- (6);
\draw (5) -- (7);
\draw (8) -- (9);
\end{tikzpicture}
\end{center}

Exercise 2: Fibonacci Heap Modifications \textit{(13 Points)}

(a) Assume that operation \texttt{decrease-key} never occurs. Show that in this case, the maximum rank $D(n)$ of a Fibonacci heap is at most $\lfloor \log_2(n) \rfloor$. \textit{(5 Points)}

(b) We want to augment the Fibonacci heap data structure by adding an operation \texttt{increase-key}(v, k) to increase the key of a node $v$ (given by a direct pointer) to the value $k$. The operation should have an amortized running time of $O(\log n)$. Describe the operation \texttt{increase-key}(v, k) in sufficient detail and prove the correctness and amortized running time. \textit{(8 Points)}

Remark: You can use the same potential function as for the standard Fibonacci heap data structure. Note however that after conducting \texttt{increase-key}(v, k) the Fibonacci heap must still be a list of heaps, where the maximum rank $D(n) \in O(\log n)$. 