1. Given a graph $G = (V, E)$ with vertex weights $w_v > 0$, we consider the problem of finding an independent set $S \subseteq V$ with the maximum total weight, i.e., $w(S) := \sum_{v \in S} w_v$ is maximized.

   Let also a *nice* tree decomposition $(Tr = (T, F), \{V_t : t \in T\})$ of G be given.

   (a) (1 point) Define for a leaf node t of Tr the maximum total weight $OPT_t(U)$ for each possible subset U of the bag V_t.

   > **Solution:** There are many ways to correctly formulate this. For example,
   >
   > 1. $OPT_t(\{v\}) = w(v)$ and $OPT_t(\emptyset) = 0$
   > 2. $OPT_t(V_t) = w(V_t)$ and $OPT_t(\emptyset) = 0$
   > 3. $OPT_t(U) = w(U)$ (where $w(\emptyset) = 0$)
   > 4. $OPT_t(U) = \sum_{v \in U} w(v)$

   (b) (1 point) Define for a forget node t of Tr the maximum total weight $OPT_t(U)$ for each possible independent subset U of the bag V_t.

   > **Solution:** Forget: V_t has one less vertex v than child t', so let's take the best out of the two possible options:
   >
   > $$OPT_t(U) = \max\{OPT_{t'}(U), OPT_{t'}(U \cup \{v\})\}$$
   >
   > - If someone confuses forget and introduce nodes, subtract $\frac{1}{2}$ only here, but not in the next question.

   (c) (2 points) Define for an introduce node t of Tr the maximum total weight $OPT_t(U)$ for each possible independent subset U of the bag V_t.

   > **Solution:** Introduce: V_t has one vertex v more than child t', so
   >
   > $$OPT_t(U) = \begin{cases} OPT_{t'}(U) & \text{if } v \notin U \\ OPT_{t'}(U \setminus \{v\}) + w(v) & \text{if } v \in U \text{ and } v \text{ has no neighbors in } U \\ -\infty & \text{if } v \in U \text{ and a neighbors of } v \text{ is in } U \end{cases}$$

   (d) (1 point) Define for a join node t of Tr the maximum total weight $OPT_t(U)$ for each possible independent subset U of the bag V_t.

   > **Solution:** Join: two children t_1 and t_2 with $V_t = V_{t_1} = V_{t_2}$, so
   >
   > $$OPT_t(U) = OPT_{t_1}(U) + OPT_{t_2}(U) - w(U)$$

   (e) (1 point) Suppose that for all tree nodes t and for all arguments U the values defined in the above recursive function have been stored in a table $M_t[U]$. Explain how to determine the maximum total weight of an independent set of G from this table.

Solution: Consider the root node r of Tr. Return the maximum over all $U \subseteq V_r$ of $M_r[U]$.

End of assignment