There are two basic ways to do this. Let $G = (V, E)$; we can give the answer without using $\mathcal{A}$ if $V = \phi$ or $k = 1$, and so we will suppose $V \neq \phi$ and $k > 1$.

The first approach is to add an extra node v^* to G, and join it to each node in V; let the resulting graph be G^*. We ask $\mathcal{A}$ whether G^* has an independent set of size at least k, and return this answer. (Note that the answer will be yes or no, since G^* is connected.) Clearly if G has an independent set of size at least k, so does G^*. But if G^* has an independent set of size at least k, then since $k > 1$, v^* will not be in this set, and so it is also an independent set in G. Thus (since we're in the case $k > 1$), G has an independent set of size at least k if and only if G^* does, and so our answer is correct. Moreover, it takes polynomial time to build G^* and ask call $\mathcal{A}$ once.

Another approach is to identify the connected components of G, using breadth-first search in time $O(|E|)$. In each connected component C, we call $\mathcal{A}$ with values $k = 1, \ldots, n$; we let k_C denote the largest value on which $\mathcal{A}$ says "yes." Thus k_C is the size of the maximum independent set in the component C. Doing this takes polynomial time per component, and hence polynomial time overall. Since nodes in different components have no edges between them, we now know that the largest independent set in G has size $\sum_C k_C$; thus, we simply compare this quantity to k.

[1] ex30.643.488