A: epsilon, 1111111111, 1#111, 11111#111#11111111111, …

An example string s0 in A: $1^p\#1^{p+1}\#1^{p+2}\#\ldots\#1^{2p-1}\#1^{2p}$

The length of xy is between one and p, and xy is made up entirely of “1” characters.
y is a suffix of xy, made up of between one and the length of xy 1s. So, the smallest version of y is 1, and the largest is 1^p.
No matter what value y has between 1 and 1^p, if we pump (repeat) it one extra time, then the resulting string s1 now has repetition. This is because the value before the first # in s1 will be a string in the range between 1^{p+1} and 1^{2p}, and all of those values are already in the string s1 somewhere after the first #.
For example, if y is 1, then s1 is $1^{p+1}\#1^{p+1}\#\ldots\#1^{2p}$, which contains a repetition.
If y is 1^2, then s1 is $1^{p+2}\#1^{p+1}\#1^{p+2}\#\ldots\#1^{2p}$, which also contains a repetition.
If y is 1^p, then s1 is $1^{2p}\#1^{p+1}\#\ldots\#1^{2p}$, which also contains a repetition.
So, I have shown that when y is the smallest length it can be, 1, pumping one additional time causes s1 to not be in A. I have also shown that when y is the largest length it can be, p, pumping one additional time causes s1 to not be in A.
Additionally, because s0 contains every string of 1s with length between p+1 and 2p, inclusive, if y is chosen to be any length between 1 and p and pumped one additional time, the resulting string s1 will have a repetition. This is because the resulting string of 1s is guaranteed to have a length between p+1 and 2p, inclusive.
Therefore, for any length of y between 1 and p, pumping one additional time causes a repetition.
So, starting with the string s0, which is in the language A, and pumping any possible length of y one additional time, we end up with a string that is not in A. Therefore, the language A is not regular.