The pumping Lemma alternates between "for every" and "there exists“ quantifiers. Notice:

1. For every regular language L
2. There exists a constant n
3. For every string w in L such that $|w| \geq n$,
4. There exists a way to break up w into three strings w = xyz such that
   - $|y| > 0$
   - $|xy| \leq n$
   - For every $k > 0$, the string xy^kz is also in L.

The pumping lemma is one tool we can use to prove that some language is not regular. Such a proof requires a proof by contradiction. A good solution makes this clear and mimics the alternating structure of the pumping lemma.

This text give an example solution that you can use to guide your own solutions on homework and exams.

Use a proof by contradiction

Show that $L=\{0^k1^k \mid k=0,1,2, \ldots\}$ is not regular.

Proof by contradiction

1. Assume that L **is regular** and show that this leads to a contradiction, thus establishing L **is not regular**.

2. If L is regular then it is recognized by some DFA, call this DFA D.

3. By the pumping lemma, D has some constant called the pumping constant, which we will call N, such that the following holds
   1. For every string w in L such that $|w| \geq N$,
   2. There exists a way to break up w into three strings w = xyz such that
      1. $|y| > 0$
      2. $|xy| \leq N$
      3. For every $k > 0$, the string xy^kz is also in L.

Restate the lemma

4. These properties are supposed to hold for every string w, so we will exhibit a single string w for which the properties don’t hold, thus a contradiction.

1. Let w = 0^N1^N , clearly |w| = 2N $\geq$ N, and 0^N1^N is recognized by L, so the preconditions hold.

Provide a string that meets the conditions

2. By the lemma we break up w into three parts: w = xyz. We need to partition the three parts across the whole of w which is 0^N1^N . Because we know
    1. |y| > 0
    2. |xy| ≤ N

The boundary between the substrings y and z must be such that xy is composed of only 0s. Visually

$0_1\ 0_2\ 0_3 \ldots 0_n\ 1_1\ 1_2\ 1_3 \ldots 1_n$

… X Y … Z

Clearly describe all possible cases

3. We must consider all possible cases Y = 0^j for j = 1..N
   |Y|=1, |Y|=2, … ,|Y|=n.

4. The third fact says if we pump y, (any number of times) then the result remains in the language L. It is easy to show this is not the case for all of the possible cases. For any j in 1 ..N, suppose Y = 0^j . Let us pump 0 times, thus the resulting string must be have j fewer 0s, but still retains all N 1s. Thus the result cannot be in the Language L since it has fewer 0s than 1s.

Argue that every case leads to a contradiction

5. This is a contradiction, so our original assumption that L is regular must be false.

6. SEE NEXT PAGE FOR MORE ADVICE!

What you get to choose and what you don’t.

a) You *don't* choose the pumping length p, it's assumed.
b) You *do* choose the string w that will be pumped.
c) You *don't* choose the breakup of the string w into xyz. you just know that |xy| <= p
d) So instead you choose the string so that x & y must contain the substring you want.
e) You *do* choose the i for part where you have xy^iz

Based on the fact that |xy| <=p you can argue that the value of i you choose ***must*** put the string outside of the language.