Pumping Lemma to prove NOT REG

1. Definition of pumping lemma: L is not in REG. To show this, we apply the pumping lemma for the class REG. Let N be the pumping lemma constant for L1. We choose the string w = ___(N一般为幂) ∈ L with |w| ≥ N, and we consider all possible factorizations w = xyz satisfying the conditions |y| ≥ 1 and |xy| ≤ N.
2. Analysis: According to the pumping lemma, the string wk =___(k一般为幂) should be in L for every k ≥ 0. case 1: since occurrences of ____ case 2:
3. Conclusion: We thus conclude that L1 is not a regular language.

Reduction

- To prove that L is not in RE, we establish a reduction Lp（已知的not RE） ≤m L.
- we need to establish a mapping m from input instances of Lp to output instances of L. - The reduction takes as input a string enc(M)（已知的） and produces as output a string enc(M1,M2, M3....)（需要证明的）, where **(根据相似的规则，构造reduction)**.
- We now show that the proposed mapping represents a valid reduction, that is： enc(M) ∈ Lp iff L(M) ___ (definition of Lp)（一般直接从题目里抄p的右边的性质） iff L(M) ____ (definition of property used for reduction)（结合reduction结果构造性质） iff L(M1) *_关系* L(M2) *_关系* L(M3)(definition of our mapping/reduction)（需要结合L构造M） iff enc(M1,M2, M3....) ∈ L (definition of L3)：（固定格式，根据L左边的M个数）

To prove Not RE

Since LP is in RE, if its complement language LP were in RE as well, then we would conclude that both languages are in REC, from a theorem in Chapter 9 of the textbook. But we have already shown that LP is not in REC. We must therefore conclude that LP is not in RE

Rice'a algorithm

We need to show that the property P is not trivial, that is, P is neither empty nor equal to RE.

1. First, ________, therefore P is not empty.
2. Second, consider the string w = , _______. The language {w} is finite and therefore also in RE. It is immediate to see that {w}/∈ P; therefore P is not equal to RE. / or not RE, but ∈ P.
3. We can now apply Rice’s theorem to conclude that, since P is not trivial, LP is not in REC.

Induction

Proof of ___, the proof is by mutual induction on the **length of x**. **base** We have |x| = 0, that is, x = ε. **Induction** Let |x| = n > 0. We can then write