CS474 - Project #1 Pseudocode
by Chandra and Hamid

Algorithm Calculate

Problem: Given a simple instruction and a block of n instructions (n < 11), identify which instructions in the block can be executed in parallel to the simple instruction.

Input: a string of simple one instruction and a list of a block of n instructions; where 0 < n < 11
Output: a string list of at least one simple instruction that can be executed in parallel to the inputted simple instruction or NONE

```python
def calculate(si, b):
  result = []
  for i in b:
    out1 = set(replace("[^a-z]", "", si[:si.index("=")]))
    in1 = set(replace("[^a-z]", "", si[si.index("="):]))
    out2 = set(replace("[^a-z]", "", i[:i.index("=")]))
    in2 = set(replace("[^a-z]", "", i[i.index("="):]))

    if !out1.intersect(in2) and
       !out2.intersect(in1) and
       !out1.intersect(out2):
         result.append(i)

     return result if len(result) > 0 else "NONE"
```

Algorithm Verify

Problem: You are given a number N < 11 instructions, identify all the pairs of instructions that can be executed in parallel.

Input: a list of n simple instruction string(s) where 1 < n < 11
Output: a list of simple instruction(s) that can b

```python
def verify(block):
  result = []

    for i in range(0, len(block) - 1):
      instructions = calculate(block[i], block[i+1:len(block)])
      if instructions != "NONE":
        for inst in instructions:
          result.append((block[i]), inst))

      return result if len(result) > 0 else "NONE"
```