**Pseudo Code**

Before we can take our pseudo-code and translate it to program code, we need to first formalize the pseudo-code a little bit more. We'll still use English, but we'll use some keywords to help us break down the program logic into concrete commands, which will then make translating to program code much easier.

We'll use the below keywords to assist us, along with their meaning.

|  |  |
| --- | --- |
| **keyword** | **meaning** |
| START | start of the program |
| SET | sets a variable we can use for later |
| GET | retrieve input from user |
| PRINT | displays output to user |
| READ | retrieve value from variable |
| IF / ELSE IF / ELSE | show conditional branches in logic |
| WHILE | show looping logic |
| END | end of the program |

#1 Sum of two integers

START

SUM two integers

SET integer1 = int

SET integer2 = int

SET sum = integer1 + integer2

Print sum

End

#2 A method that takes and array of strings, and returns a new array with every other element of the original array starting with the first element

START

GET list\_of\_words = [‘list’, ‘of’, ‘words’, ‘split’, ‘into’, ‘individual’]

SET words = ‘’

For each word in list\_of\_worlds

word += word

print list\_of\_words

END

#3 a method that takes an array of integers, and returns a new array with every other element from the original array, starting with the first element.

START

GET arr = [1, 2, 3, 4, 5 ]

SET odd\_arr = []

For each number in arr

If the index is odd

Add the number to the odd array

Print the odd\_arr

END

#3 a method that determines the index of the 3rd occurrence of a given character in a string. For instance, if the given character is 'x' and the string is 'axbxcdxex', the method should return 6 (the index of the 3rd 'x'). If the given character does not occur at least 3 times, return nil

START

GET string = ‘abbabbabbabbaabb’

GET search\_letter = ‘a’

For each letter in the string

If letter is the same as the search letter

Count += 1

If count equals 3

PRINT index of letter

ELSE IF count < 3

PRINT nil

END

#4 a method that takes two arrays of numbers and returns the result of merging the arrays. The elements of the first array should become the elements at the even indexes of the returned array, while the elements of the second array should become the elements at the odd indexes. For instance:

START

GET arr1 = [1, 2, 3]

GET arr2 = [4, 5, 6]

SET merged\_arr = []

For each char in arr1 and arr2

Copy to merged\_arr

END