```
Script started on Thu 20 Mar 2014 09:43:07 AM EDT
cab35@hoare:~/Documents/cs214/labs/lab06$ cat ac␇␈␛[Kv␇erage.cpp
/* average.cpp "test-drives" function Average.
 * Precondition: theArray is an array of numbers.
 * Output: the average of the values.
 *
 * Begun by: Dr. Adams, for CS 214 at Calvin College.
 * Completed by:Charles Blum
 * Date: 20 March 2014
 */

#include <iostream>
#include <numeric>
using namespace std;

double average(double anArray[], int itsSize);

int main() {

  // Define array0 as an array containing no values
  double array0[] = {};
  // Define array1 as an array containing 9.0, 8.0, 7.0, 6.0
  double array1[] = { 6.0, 7.0, 8.0, 9.0 };

  cout << "\n average 1 is " << average(array0, 0) << endl;
  cout << "\n average 2 is " << average(array1, 4) << endl;
}

/*******************************************************
 * average averages the values in an array of doubles. *
 * Receive: anArray, an array of doubles.              *
 * Return: the average of the values in aVector.       *
 *******************************************************/
double average(double a[], int size) {
  if ( size == 0 )
    return 0.0;
  else
    return accumulate( a, a+size, 0.0 ) / size;
}
cab35@hoare:~/Documents/cs214/labs/lab06$ ./cpp_average

 average 1 is 0

 average 2 is 7.5
cab35@hoare:~/Documents/cs214/labs/lab06$ exit

Script done on Thu 20 Mar 2014 09:43:20 AM EDT
Script started on Thu 20 Mar 2014 10:04:29 AM EDT
cab35@hoare:~/Documents/cs214/labs/lab06$ cat av␇erage.adb
-- average.adb "test-drives" function Average.
-- Precondition: theArray is an array of numbers.
-- Output: the average of the numbers.
--
-- Begun by: Dr. Adams, CS 214 at Calvin College.
-- Completed by:Charles Blum
-- Date: 20 March 2014
-----------------------------------------------

with Ada.Text_IO; use Ada.Text_IO;             -- Put(String)
with Ada.Float_Text_IO; use Ada.Float_Text_IO; -- Put(Float)

procedure average is

-- Declare Vector type
type Vector is array(Positive range <>) of Float;
-- Define array0 as an array containing no values
array0 : Vector := (0.0, 0.0);
```

```
-- Define array1 as an array containing 9, 8, 7, 6
array1 : Vector := (6.0, 7.0, 8.0, 9.0);

----------------------------------------------
-- sum() sums the values in an array           -
-- Receive: anArray, an array of numbers     -
-- Return: the sum of the values in anArray. -
----------------------------------------------

function sum( a: Vector ) return Float is
Total : Float := 0.0;
begin
  for I in A'Range
  loop
    Total := Total + A(I);
  end loop;
  return Total;
end sum;

function computeAverage( a: Vector ) return Float is
Total : Float := 0.0;
begin
  Total := sum(a);
  return Total / Float( a'Length );


end computeAverage;

begin
   Put(" average 0 is ");
   Put( computeAverage(array0) );
   New_line;
   Put(" average 1 is ");
   Put( computeAverage(array1) );
   New_line;
end average;

cab35@hoare:~/Documents/cs214/labs/lab06$ ./average
 average 0 is  0.00000E+00
 average 1 is  7.50000E+00
cab35@hoare:~/Documents/cs214/labs/lab06$ exit

Script done on Thu 20 Mar 2014 10:04:38 AM EDT
Script started on Sun 30 Mar 2014 07:06:05 PM EDT
theo@theo-VBox:~/Desktop/cs214/lab06$ cat av␇erage.el
;;; average.el "test-drives" function average().
;;; Output: the average of a sequence of numbers.
;;;
;;; Begun by: Dr. Adams, CS 214 at Calvin College.
;;; Completed by: Charles Blum
;;; Date: 30 March 2014
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; sum() sums the values in an array.       ;;
;;; Receive: anArray, an array of numbers,   ;;
;;;          itsSize, its size.              ;;
;;; Return: the sum of the values in anArray.;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
(defun sum (anArray itsSize)
  (if (vectorp anArray)
      (if (<= itsSize 0)
          0.0
        (+ (aref anArray (- itsSize 1)) (sum anArray (- itsSize 1)))
      )
    nil)
)
```

```
sum

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; average() computes the average of an array of numbers.;;
;;; Receive: anArray, an array of numbers.              ;;
;;; Return: the average of the numbers.                 ;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
(defun average (anArray)
  (if (vectorp anArray)
      (if (<= (length anArray) 0)
          0.0
        (/ (sum anArray (length anArray)) (length anArray))
        )
    nil)
)
average

;;; Test function sum
(sum [] 0)
0.0

(sum [9.0 8.0 7.0 6.0] 4)
30.0

;;; Test function average
(average [])
0.0

(average [9.0 8.0 7.0 6.0])
7.5



theo@theo-VBox:~/Desktop/cs214/lab06$ exit

Script done on Sun 30 Mar 2014 07:06:16 PM EDT
Script started on Sun 30 Mar 2014 06:39:30 PM EDT
theo@theo-VBox:~/Desktop/cs214/lab06$ cat av␇erage.rb
# average.rb averages the values of an array of doubles.
# Precondition: theArray is an array of numbers
# Output: the average of the values in theArray
#
# Begun by: Dr. Adams, for CS 214 at Calvin College.
# Completed by:: Charles Blum
# Date: 30 March 2014
########################################################

###############################################
# sum() sums the values in an array
# Receive: anArray, an array of numbers
# Return: the sum of the values in anArray.
################################################

# Define sum() method here
def sum(someArray)
  total = 0.0
  someArray.each { |a| total += a }
  total
end

# Document and define average() method here
def average(someArray)
  if someArray.size <= 0
    average = 0.0
  else
    average = sum(someArray) / someArray.size
  end
end

def main
  # Define array0 as an Array containing no values
  array0 = []
  # Define array1 as an Array containing 9.0, 8.0, 7.0, 6.0
  array1 = [9.0, 8.0, 7.0, 6.0]

  puts "sum 0 is: #{ sum(array0) }\n"
  puts "sum 1 is: #{ sum(array1) }\n"

  puts "average 0 is: #{ average(array0) }\n"
  puts "average 1 is: #{ average(array1) }\n"
end

main

theo@theo-VBox:~/Desktop/cs214/lab06$ ruby aver␇age.rb
sum 0 is: 0.0
sum 1 is: 30.0
average 0 is: 0.0
average 1 is: 7.5
theo@theo-VBox:~/Desktop/cs214/lab06$ exit

Script done on Sun 30 Mar 2014 06:39:46 PM EDT
```