

R-4.9:

example 1

Code Fragment 4.12 is calculating the sum of all integers in the given array by looping through each element
 $\Rightarrow O(n)$

R-4.10:

Code Fragment 4.12 example 2 is calculating the sum of all even index number position integers in the given array by looping through it
 $\Rightarrow O(\frac{1}{2}n) = O(n)$

R-4.11:

Code Fragment 4.12 example 3 is calculating the sum of all integers in the given array using 2 loops. (nested loop).

$$f(n) = \sum_{i=arr[0]}^{arr[n]} \text{total} + \sum_{i=array[0]}^{arr[n-1]} \text{total} + \dots + \sum_{i=arr[0]}^{arr[0]} \text{total}$$

Because it uses 2 loops $\Rightarrow O(n^2)$

R-4.12:

Code Fragment 4.12 example 4 returns the sum of prefix sums
 $\Rightarrow O(n)$

R-4.13:

Code Fragment 4.12 example 5 returns the number of times the second array stores sum of prefix sums from the first array.

$$\Rightarrow O(n^3)$$
