

Lesson 6 Wrap-up

Write a function called **max_sum** that takes **v**, a row vector of numbers, and **n**, a positive integer as inputs. The function needs to find the **n** consecutive elements of **v** whose sum is the largest possible. In other words, if **v** is **[1 2 3 4 5 4 3 2 1]** and **n** is 3, it will find 4 5 and 4 because their sum of 13 is the largest of any 3 consecutive elements of **v**. If multiple such sequences exist in **v**, **max_sum** returns the first one. The function returns **summa**, the sum as the first output argument and **index**, the index of the first element of the **n** consecutive ones as the second output. If the input **n** is larger than the number of elements of **v**, the function returns 0 as the sum and -1 as the index. Here are a few example runs:

```
[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],3)
    summa = 13
    index = 4
[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],2)
    summa = 9
    index = 4
[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],1)
    summa = 5
    index = 5
[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],9)
    summa = 25
    index = 1
[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],10)
    summa = 0
    index = -1
```

Your Function

 Save  Reset  MATLAB Documentation (<https://www.mathworks.com/help/>)

1

Code to call your function

 Reset

1

[summa, index] = max_sum([1 2 3 4 5 4 3 2 1],3)

 Run Function 

Assessment:

Submit 

[1 2 3 4 5 4 3 2 1]

random vectors