6.If the numbers to are written out in words: one, two, three, four, five, then there are 3 + 3 + 5 + 4 + 4 = 19 letters used in total.

If all the numbers from 1 to 1000 (one thousand) inclusive were written out in words, how many letters would be used?

NOTE: Do not count spaces or hyphens. For example, 342 (three hundred and forty-two) contains 23 letters and 115 (one hundred and fifteen contains 20 letters. The use of "and" when writing out numbers is in compliance with British usage.

#6

```
def numbers(n):
  ones=["","one","two","three","four","five","six","seven","eight","nine","ten","eleven","twelve","thirteen","founteen","fifteen","sixteen","seventeen","ei
  tens=["","","twenty","thirty","fourty","fifty","sixty","seventy","eighty","ninety"]
  thounsands=["","thousand"]
  if n==0:
    return "zero"
  if n<20:
    return ones[n]
  elif n<100:
    return tens[n//10]+ones[n%10]
  elif n<1000:
    if n%100==0:
      return ones[n//100]+"hundred"
    else:
      return ones[n//100]+"hundredand"+numbers(n%100)
  else:
    return "onethousand"
total_words=sum(len(numbers(i)) for i in range (1,1001))
print(total_words)
```

```
21224
```

7. There are several cards arranged in a row, and each card has an associated number of points. The points are given in the integer array cardPoints.

In one step, you can take one card from the beginning or from the end of the row. You have to take exactly k cards.

Your score is the sum of the points of the cards you have taken.

Given the integer array cardPoints and the integer k, return the maximum score you can obtain.

Example 1: Input: cardPoints = [1,2,3,4,5,6,1], k = 3 Output: 12

Example 2: Input: cardPoints = [2,2,2], k = 2 Output: 4

Example 3: Input: cardPoints = [9,7,7,9,7,7,9], k = 7 Output: 55

#7

```
def max(cardPoints,k):
  n=len(cardPoints)
  total_sum=sum(cardPoints)
  window_size=n-k
  window_sum=sum(cardPoints[:window_size])
  min_window_sum=window_sum

  for i in range(1,n-window_size+1):
    window_sum=window_sum-cardPoints[i-1]+cardPoints[i+window_size-1]
    min_window_sum=min(min_window_sum,window_sum)
  return total_sum-min_window_sum

#getting input from user
cardPoints=list(map(int,input().strip().split()))
k=int(input())
#provide n spaced integers for the array
print(max(cardPoints,k))
```

```
9 7 7 9 7 7 9
7
55
```