**Making Change**

Assume we have a set of N>0 tokens each of different value $\{v_1, v_2, \ldots, v_N\}$ with an unlimited supply of each token from which we have to give a customer some change.

What is the minimum number of tokens to return as change?

Write a function
`int mintokens(int c, int a[], int len)`
where
`c` is the amount of change owed to the customer
`a[]` is a 1-dimensonal array of token values
`len` is the number of elements (i.e., number of different token values) in `a[]`
and returns the minimum number of tokens it takes to return change due `c`
or -1 if change is not possible or if `len,c`<1

File you must submit: `soln_func.cc`

*Examples:*

c = 30 a[] = {5, 10, 25}
Returns: 2
*Explanation: 1x5 + 1x25*

c = 11 a[] = {5, 9, 1, 6}
Returns: 2
*Explanation: 1x5 + 1x6*

c = 15 a[] = {3, 17, 1, 21, 4}
Returns: 4
*Explanation: 1x3 + 3x4*

c = 7 a[] = {5, 9, 3, 6}
Returns: -1
*Explanation: Not possible to make change.*

c = 7 a[] = {}
c = 0, a[]={1, 2}
Each returns: -1
*Explanation: For each, at least one of the following is true; `len,c`<1*