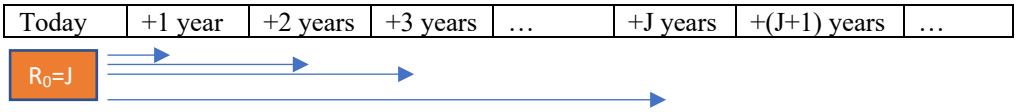
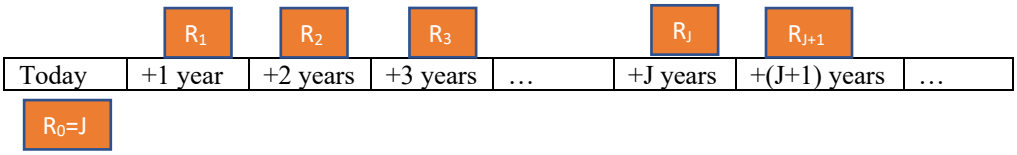
Time Traveler

You are a time traveler who has a time machine that may be only used once and whose range (R_0) allows you to jump forward in time up to J years (i.e., it can move you forward in time 1 year, 2 years, …, or J years).



In each future year, there is another time machine that may also only be used once to move you forward in time up to some number of years. The range of each future time machine may be different. You have a "time map" that specifies the range of each future time machine for a finite number of years into the future.



What is the minimum number of time machines you can use to move forward at least N years?

Write a function
`int mintimesteps(int a[], int len, int n)`
where
`a[]` is a 1D array where `a[i]` reports the range of the time machine `i` years in the future (i.e., your "time map")
`len` is the number of elements in `a[]`
`n` is the minimum number of years you want to move forward in time
and returns the minimum number of time machines needed to travel at least `n` years into the future,
otherwise returns -1 if it is not possible based on "time map" (i.e., `a[]`)

File you must submit: `soln_func.cc`

Examples:

a={2, 1, 3} n=4
Returns: 2
Explanation: Your first time machine (a[0]) allows you to move forward either 1 or 2 years.
If you choose to move forward only 1 year, then the next time machine (a[1]) allows you to move forward only 1 year (to a[2]). That year's time machine allows you to move forward 1, 2, or 3 years. Moving forward either 2 or 3 years will bring you to your time destination >=4 years.
*In this case, you would have used **three** time machines: 1->1->2 (or 1->1->3)*
*Instead, if in the first year you moved forward 2 years (to a[2]), then you can complete your journey by using just one more time machine for a total of **two** time machines: 2->2 (or 2->3).*

a={1, 1, 1} n=3
Returns: 3
Explanation: Only one possible sequence; 1->1->1

a={1, 1, 1, 1} n=10
Returns: -1
Explanation: Not possible to move forward >=10 years with this time map.

a={1, 2, 3, 1, 2, 4} n=7
Returns: 4
Explanation: 1->1->3->2 or 1->1->3->3 or 1->1->3->4.