Chapter 6: Exercise 2 11912039 郑鑫颖.

The problem. Given sets of values $\ell_1, \ell_2, \ldots, \ell_n$ and $h_1, h_2, \ldots, h_n$, find a plan of maximum value. (Such a plan will be called *optimal*.)

Example. Suppose $n = 4$, and the values of ℓ_i and h_i are given by the following table. Then the plan of maximum value would be to choose "none" in week 1, a high-stress job in week 2, and low-stress jobs in weeks 3 and 4. The value of this plan would be $0 + 50 + 10 + 10 = 70$.

	Week 1	Week 2	Week 3	Week 4
ℓ	10	1	10	10
h	5	50	5	1

(a) Show that the following algorithm does not correctly solve this problem, by giving an instance on which it does not return the correct answer.

```
For iterations i = 1 to n
  If h_{i+1} > ℓ_i + ℓ_{i+1} then
    Output "Choose no job in week i"
    Output "Choose a high-stress job in week i+1"
    Continue with iteration i+2
  Else
    Output "Choose a low-stress job in week i"
    Continue with iteration i+1
  Endif
End
```

To avoid problems with overflowing array bounds, we define $h_i = \ell_i = 0$ when $i > n$.

In your example, say what the correct answer is and also what the above algorithm finds.

Week	1	2	3	4
l	10	1	10	10
h	50	5	5	1

for 1 to 4

1: choose 10 in week 1

2: choose 1 in week 2

3: choose 10 in week 3

4: choose 10 in week 4

$10+1+10+10=31$

is not optimal

correct is: $50+1+10+10=71$

(b) Give an efficient algorithm that takes values for $\ell_1, \ell_2, \ldots, \ell_n$ and $h_1, h_2, \ldots, h_n$ and returns the *value* of an optimal plan.

define. op(i): the optimal choice from week 1 to week i.

$$op(i) = \begin{cases} 0, & i \le 0 \\ \max(op(i-1) + l_i,\ op(i-2) + h_i) \end{cases}$$

pseudo code

```
for i = -1 to n
    if i ≤ 0 op(i) = 0
    else op(i) = max(op(i-1) + li, op(i-2) + hi)
```

time complexity $O(n)$