Introduction to Algorithms Lecture 8

By: Darmen Kariboz

Dynamic programming. Part II

- Longest palindrome
- Goat
- Stones and knapsack
- Gold Mine
- Cutting a Rod

Longest palindrome

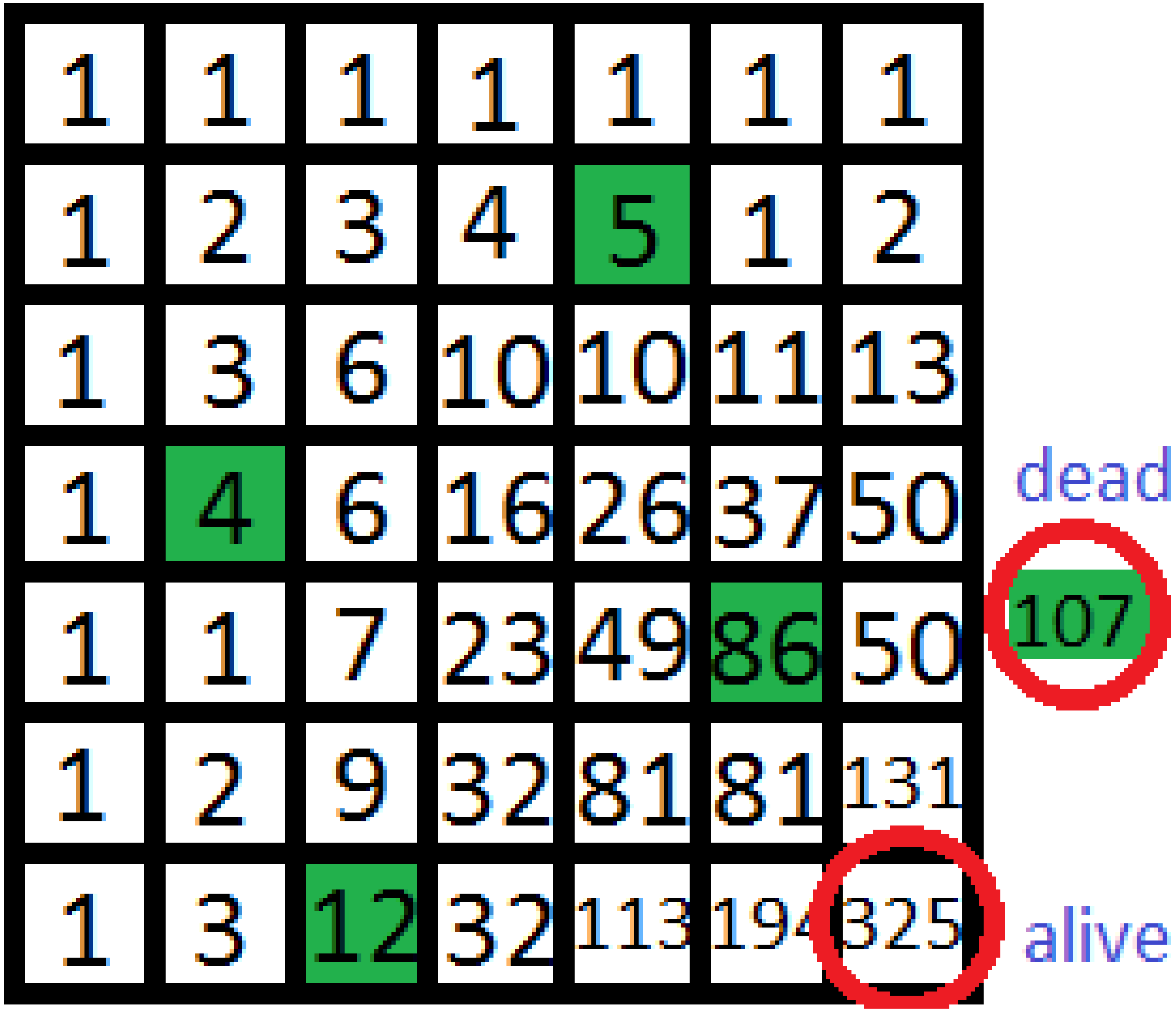
Given string, find longest possible palindrome, that can be found from given string by removing some letters.

Solution

- If given S = ‘pmadatm’, largest palindrome is ‘madam’.
- Reverse given word K = ‘mtadamp’.
- And find largest substring between S and K.
- Result is ‘madam’!

Goat

We have a map NxM.
Goat is on cell 1x1. Can go only to right and down.
Some places have swamp so goat will die if step there.
Goat’s home is in cell NxM
Question: How many ways for goat to die, and how many to go home *alive*.



1	1	1	1	1	1	1	
1	2	3	4	5	1	2	
1	3	6	10	10	11	13	
1	4	6	16	26	37	50	dead
1	1	7	23	49	86	50	107
1	2	9	32	81	81	131	
1	3	12	32	113	194	325	alive

Goat

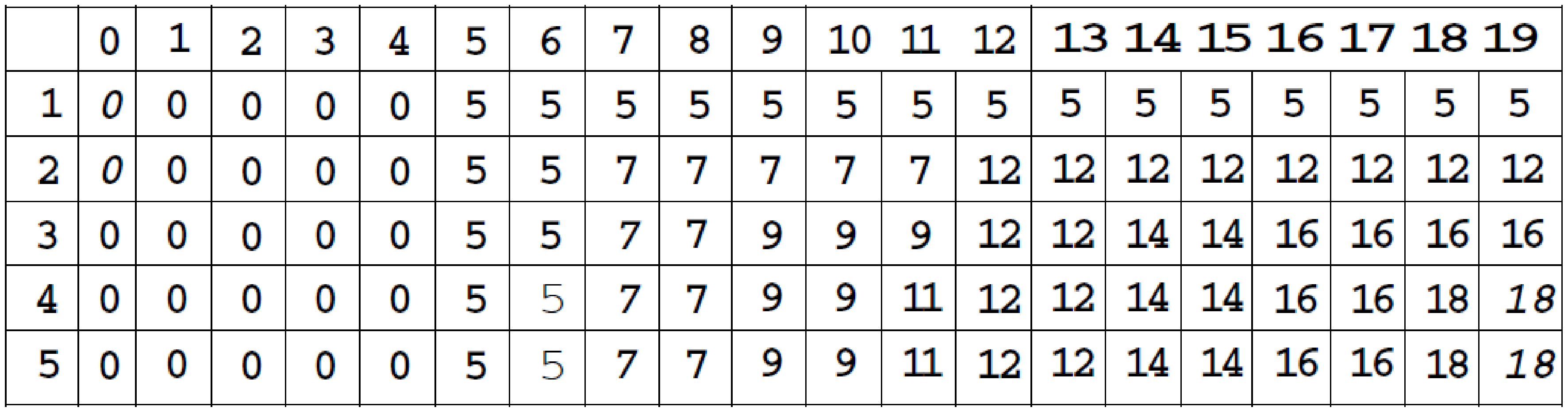
```
For i = 1  To N Do
    For j = 1 To M Do
        //check that coordinates within map
        ways = A[i-1][j]+A[i][j-1]
        If (i,j) is swamp then
            dead = dead + ways
        else
            A[i][j] = ways
Ouput dead and A[N][M]
```

Stone and knapsack

Given N stones with some weight, and knapsack that can take maximum W weight units. What is maximum weight you can put into knapsack by use of these stones.

EX: N=5; W=19

$p_1 = 5$; $p_2 = 7$; $p_3 = 9$; $p_4 = 11$; $p_5 = 13$



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	*0*	0	0	0	0	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
2	*0*	0	0	0	0	5	5	7	7	7	7	7	12	12	12	12	12	12	12	12
3	0	0	0	0	0	5	5	7	7	9	9	9	12	12	14	14	16	16	16	16
4	0	0	0	0	0	5	5	7	7	9	9	11	12	12	14	14	16	16	18	*18*
5	0	0	0	0	0	5	5	7	7	9	9	11	12	12	14	14	16	16	18	*18*

```
• For i:=2 To N Do
• For j:=1 To W Do Begin
•     If j-P[i]>=0 Then
•         A[i,j]:= Max (A[i-l ,j] , A[ i-l,j-P[i] ]+P[i])
•     Else A[i,j]:=A[i-l,j];
• End;
```

Gold Mine Problem

Given a gold mine of n*m dimensions. Initially the miner is at first column but can be at any row. He can move only (right->,right up /,right down\)

Find out maximum amount of gold he can collect.

```
For i:=1 To N Do
For j:=2 To M Do Begin
    int right = a[i, j-1]
      // check if we can go up
    int rightUp = a[i-1, j-1]
      // check if we can go down
    int rightDowm = a[i+1, j-1]
    a[i , j] = a[i, j] + max(right, rightUp, rightDown)
End;
```

Length	1	2	3	4	5	6	7	8
Price	1	5	8	9	10	17	17	20

Cutting a Rod

Given a rod of length n inches and an array of prices that contains prices of all pieces of size smaller than n. Determine the maximum value obtainable by cutting up the rod and selling the pieces.

```
Create array val[];
val[0] = 0;

for i = 1 -> n+1 ; do
    max_val = Integer.MIN_VALUE;
    for j = 0 -> i; do
        max_val = max(max_val, price[j] + val[i-j-1]);
        val[i] = max_val;
    }

return val[n];
```

Home Work

- **Realize Knapsack problem**
- **Realize Gold Mine problem**