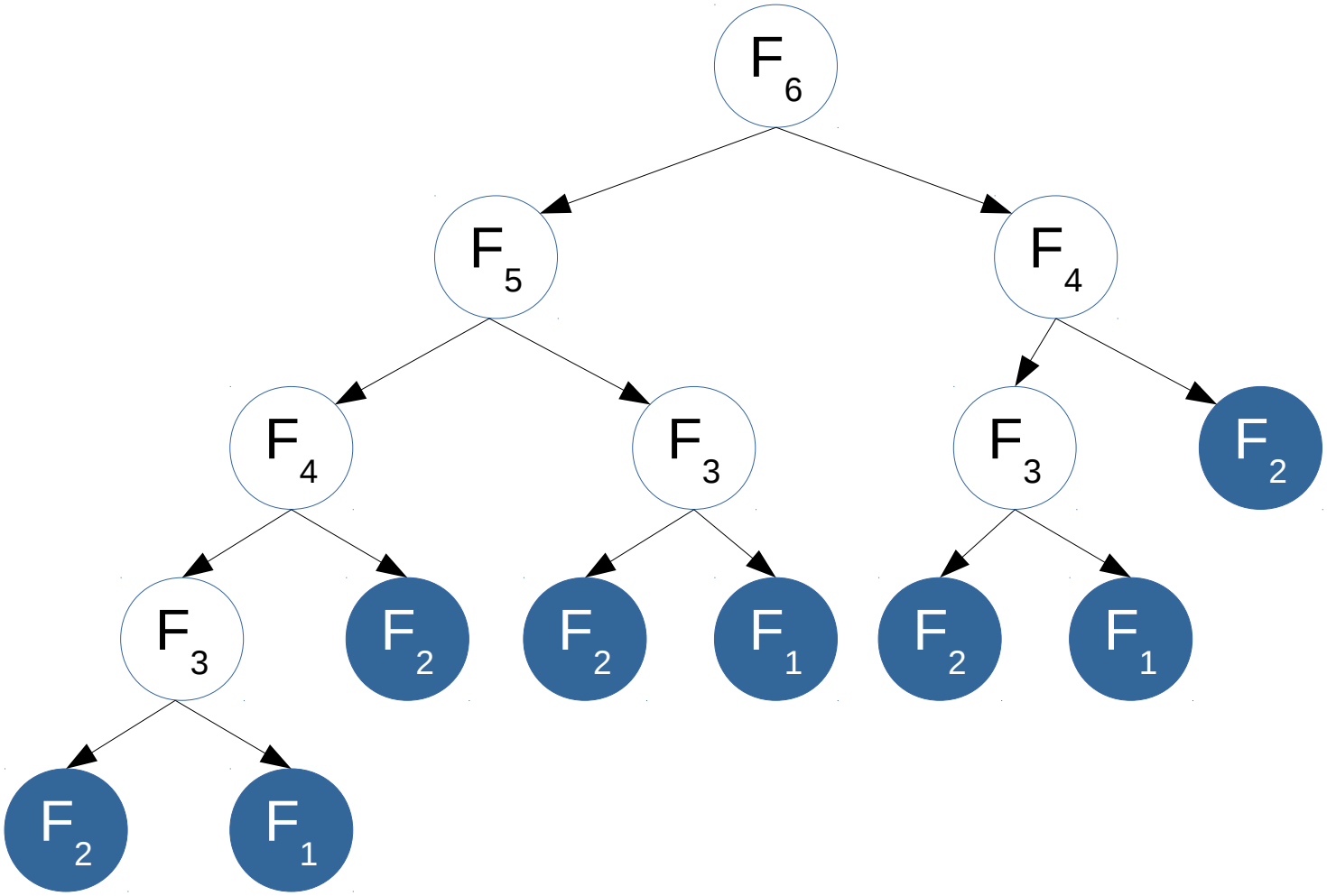
Programowanie dynamiczne

Ciąg Fibonacciego

$$F_n := \begin{cases} 0 & \text{dla } n = 0; \\ 1 & \text{dla } n = 1; \\ F_{n-1} + F_{n-2} & \text{dla } n > 1. \end{cases}$$

Programowanie dynamiczne

Ciąg Fibonacciego rekurencyjnie



Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

	0	1	2	3	4	5	6	7	...	n
i=3	0	**1**	**1**	2	-	-	-	-	...	-

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

	0	1	2	3	4	5	6	7	...	n
i=4	0	1	**1**	**2**	3	**-**	**-**	**-**	**...**	**-**

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

	0	1	2	3	4	5	6	7	...	n
i=5	0	1	1	**2**	**3**	5	-	-	...	-

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

	0	1	2	3	4	5	6	7	...	n
i=6	0	1	1	2	**3**	**5**	8	-	...	-

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

	0	1	2	3	4	5	6	7	...	n
i=7	0	1	1	2	3	**5**	**8**	13	...	-

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie

(1) i=3; t[]=

0	1	2	3	4	5	6	7	...	n
0	1	1	-	-	-	-	-	...	-

(2) dopóki (i<=n):

(3) t[i]=t[i-1]+t[i-2]

(4) i=i+1

(5) wynikiem jest t[n]

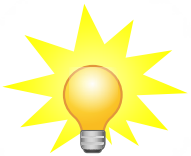
	0	1	2	3	4	5	6	7	...	n
i=n	0	1	1	2	3	5	8	13	...	F_n

Programowanie dynamiczne

Ciąg Fibonacciego dynamicznie lepiej

Czy można to zrobić lepiej?

Złożoność czasowa zmniejszyła się do O(n), ale pamięciowa wynosi także O(n)



Korzystamy z tylko dwóch poprzednich pól tablicy

Zatem tablicę mogą zastąpić trzy zmienne i złożoność pamięciowa wyniesie O(1)

```
(1) i=3; a=1; b=1; c=1;
(2) dopóki (i<=n):
        (3) c=a+b
        (4) a=b
        (5) b=c
        (6) i=i+1
(7) wynikiem jest c
```

Programowanie dynamiczne

Truskawkowe żniwa

40	38	35	30	35	16
11	7	20	18	1	6
34	2	25	9	3	28
0	23	15	16	9	17
21	37	2	34	6	1

Programowanie dynamiczne

Truskawkowe żniwa

0	0	0	0	0	0	0
0	40	38	35	30	35	16
0	11	7	20	18	1	6
0	34	2	25	9	3	28
0	0	23	15	16	9	17
0	21	37	2	34	6	1

Programowanie dynamiczne

Truskawkowe żniwa

0	0	0	0	0	0	0
0	40	38	35	30	35	16
0	11	7	**20**	18	1	6
0	34	**2**	25	9	3	28
0	0	23	15	16	9	17
0	21	37	2	34	6	1

$$T[i][j]=T[i][j]+\max(T[i-1][j],T[i][j-1])$$