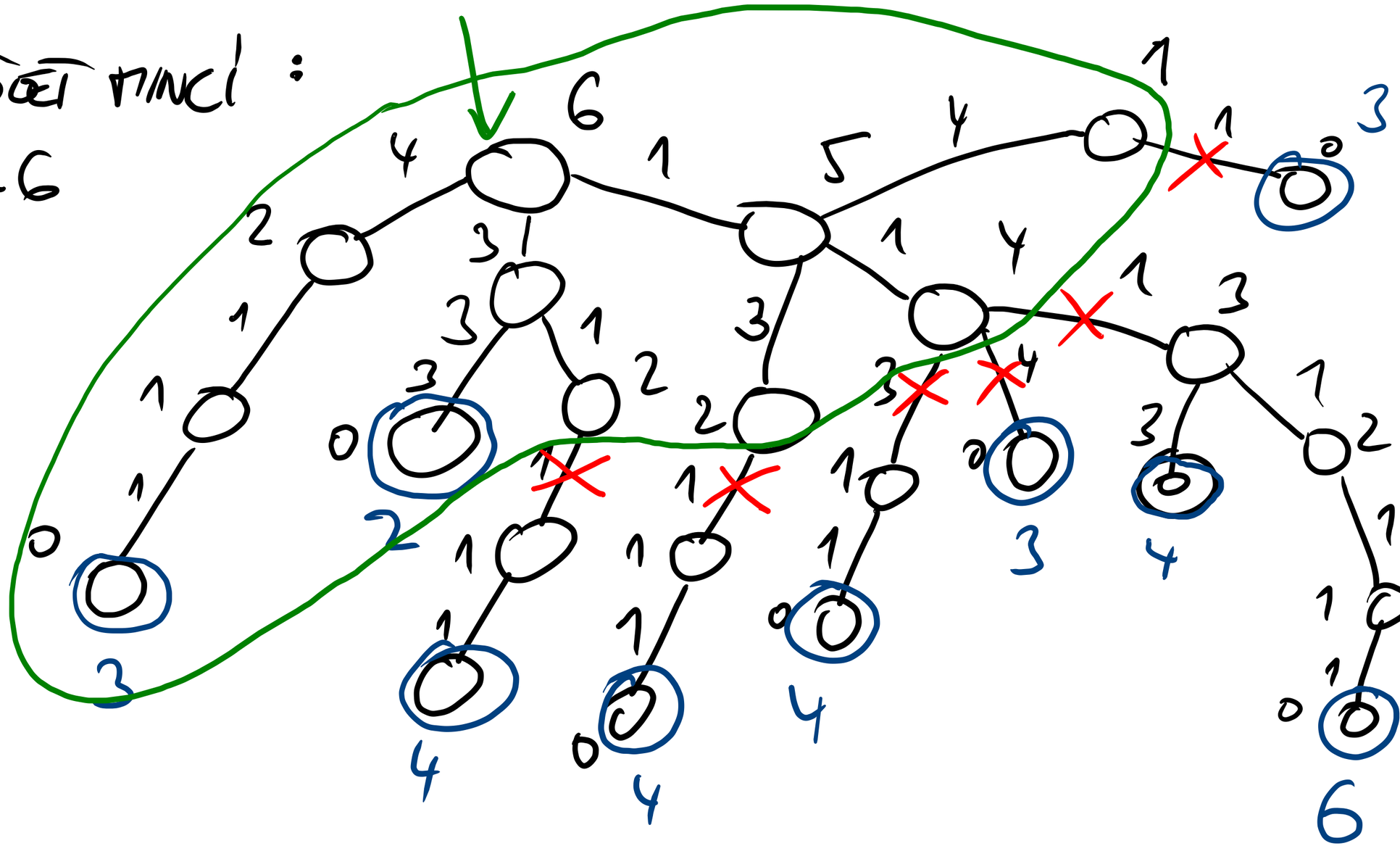
BACKTRACKING

MINIMÁLNÍ POČET MINCÍ:

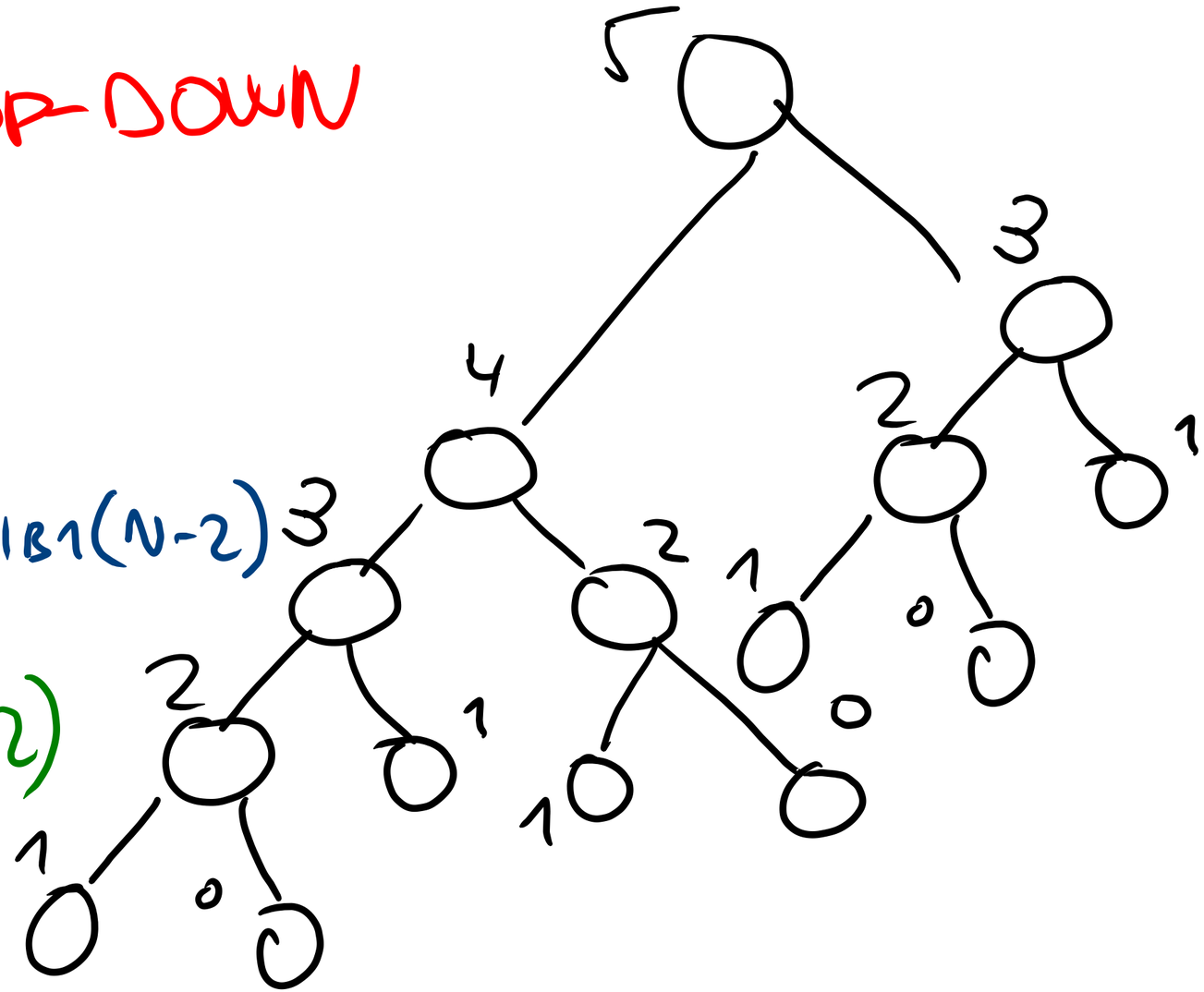
hodnoty 1, 3, 4 částka 6



FIB. POSLOUPNOST

TOP-DOWN

```
DEF FIB1(N):
    IF N == 0:
        RETURN 0
    IF N == 1:
        RETURN 1
    RETURN FIB1(N-1) + FIB1(N-2)
```



$T(n) = T(n-1) + T(n-2)$

$< 2T(n-1)$

$\in O(2^n)$

TOP-DOWN + CACHE

```
CACHE = {0: 0, 1: 1}

DEF FIB2(N):
    IF N IN CACHE:
        RETUEN CACHE[N]
    ELSE:
        X = FIB2(N-1) + FIB2(N-2)
        CACHE[N] = X
        RETURN X
```

MODUL FUNCTOOLS

```
@LRU_CACHE
DEF FIB1(N):
    .
    .
    .
```

LRU = LEAST RECENTLY USED

BOTTOM-UP

```
DEF FIB3(N):
    CACHE = {0:0, 1:1}
    FOR I=2 TO N:
        CACHE[I] = CACHE[I-1] + CACHE[I-2]
    RETURN CACHE[N]
```

SOLUTOSI: O(N)

PROBLÉM BATOHU (KNAPSACK)

BATOH → OMEZENÍ KAPACITA (W = kapacita (kg))

PŘEDMĚTY → CENA v_i
→ HMOTNOST w_i

CÍL: MAXIMALIZACE HODNOTY PŘEDMĚTŮ V BATOHU
ZA DODRŽENÍ OMEZENÍ HMOTNOSTI

i	v_i	w_i
1	1	1
2	6	2
3	18	5
4	22	6
5	28	7

$W = 11$

$OPT(5, 11) = ?$

optimalna cena

$$OPT(i, w) = \begin{cases} 0 & i = 0 \\ OPT(i-1, w) & w_i > w \\ \max\{ OPT(i-1, w), \; v_i + OPT(i-1, w-w_i) \} & \end{cases}$$

max. cena predmeta

trenutna kapacita

KNAPSACK $(n, W, w_1, \ldots, w_n, v_1, \ldots, v_n)$

FOR $w = 0$ TO W

  $M[0, w] \leftarrow 0$.

FOR $i = 1$ TO n

  FOR $w = 1$ TO W

  IF $(w_i > w)$ $M[i, w] \leftarrow M[i-1, w]$.

  ELSE $M[i, w] \leftarrow \max \{ M[i-1, w],\ v_i + M[i-1, w-w_i] \}$.

RETURN $M[n, W]$.

BOTTOM-UP

PREDMETY

ZMENŠUJÍCÍ

KAPACITA

$$OPT(i,w) = \begin{cases} 0 & \text{if } i = 0 \\ OPT(i-1,w) & \text{if } w_i > w \\ \max\{ OPT(i-1,w),\ v_i + OPT(i-1,w-w_i) \} & \text{otherwise} \end{cases}$$

TOP-DOWN

Knapsack algorithm demo

i	v_i	w_i
1	1	1
2	6	2
3	18	5
4	22	6
5	28	7

$$OPT(i,w) = \begin{cases} 0 & \text{if } i = 0 \\ OPT(i-1,w) & \text{if } w_i > w \\ \max\{\, OPT(i-1,w),\ \ v_i + OPT(i-1,w-w_i)\,\} & \text{otherwise} \end{cases}$$

weight limit w

subset of items 1, …, i	0	1	2	3	4	5	6	7	8	9	10	11
{ }	0	0	0	0	0	0	0	0	0	0	0	0
{ 1 }	0	1	1	1	1	1	1	1	1	1	1	1
{ 1, 2 }	0	1	6	7	7	7	7	7	7	7	7	7
{ 1, 2, 3 }	0	1	6	7	7	18	19	24	25	25	25	25
{ 1, 2, 3, 4 }	0	1	6	7	7	18	22	24	28	29	29	40
{ 1, 2, 3, 4, 5 }	0	1	6	7	7	18	22	28	29	34	35	40

OPT(i, w) = max profit subset of items 1, …, i with weight limit w.

SLOŽITOST: $O(n \cdot w)$

$OPT(5,12) =$

$MAX\{ OPT(1,2),$

$6 + OPT(1,0) \quad 3$

$= OPT(5,11)$