

## Řešení domácích úkolů - 4. týden

# Pascalův trojúhelník

Načtete ze standardního vstupu přirozené číslo  $n = 1, 2, \dots$

a na standardní výstup vytisknete seznam seznamů, obsahující prvních  $n$  řádek

Pascalova trojúhelníku.

1. Použijte rekurenční relace mezi následujícími řádky Pascalova trojúhelníku.
2. Řešení, která budou obsahovat výpočet kombinačních čísel anebo faktoriálů, anebo budou volat příslušné knihovní funkce, nebudou uznána.
3. Výsledný seznam vytisknete najednou a celý, včetně hranatých závorek.

## Příklad

### Vstup:

5

### Výstup:

```
[[1], [1, 1], [1, 2, 1], [1, 3, 3, 1], [1, 4, 6, 4, 1]]
```

## Řešení

**Analýza** Toto je jednoduchá úloha, stačí jenom nějak šikovně realizovat rekurzi. Nakonec tiskneme seznam seznamů (=řádků) i s hranatými závorkami.

## Vzorové řešení

Následující řádek dopočítáváme tak, že z posledního řádku vytvoříme dva seznamy tak, že přidáme nulu na začátek, resp. na konec, a oba řádky sečteme.

```
1 n = int(input())
2 pascal = [[1]]
3 for i in range(2,n+1):
4     newrow = [0] + pascal[-1]
5     newrow2 = pascal[-1] + [0]
6     for j in range(len(newrow)):
7         newrow[j] += newrow2[j]
8     pascal.append(newrow)
9 print(pascal)
```

## Alternativní řešení

Sečtení dvou seznamů po členech jde vyjádřit i kompaktněji:

```
1 n = int(input())
2 pascal = [[1]]
3 for _ in range(1,n):
4     newrow = [i + j for i, j in zip([0] + pascal[-1], pascal[-1] + [0])]
5     pascal.append(newrow)
6 print(pascal)
```

Tady se využívá funkce `zip`, která spojuje odpovídající prvky argumentů do n-tic a vrátí seznam těchto n-tic.

V této verzi také píšeme namísto proměnné cyklu i nepojmenovanou proměnnou `_`, abychom zdůraznili, že proměnná cyklu se nikde v cyklu nepoužívá.