

# Problémy - zadání

---

## Fibonacciho posloupnost

---

Napište funkci `int fib(int n)`. Funkce by měla vrátit  $n$ -té fibonacciho číslo.

`fib(1) = 1`

`fib(2) = 1`

Pro vygenerování dalšího fib. čísla sečteme předchozí dvě.

Příklad:

n:    1 2 3 4 5 6 7 8 9

fib(n) 1 1 2 3 5 8 13 21 34

## Tabulkový cestovatel

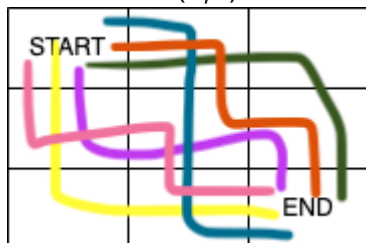
---

Máte cestovatele v 2D tabulce. Cestovatel začíná v levém horním rohu a cestuje do pozice v pravém dolním rohu. Může se pohnout jen doprava a dolu.

Funkce `int GridTraveler(int m, int n)` zpočítá kolika různými způsoby lze cestu provést?

Příklad:

`GridTraveler(2,3) = 6`



## Možný součet (CanSum)

---

Funkce `bool CanSum(int soucet, int [] cisla)` vrácí boolean, který indikuje, zda lze z čísel zadaných v argumentu `cisla` vytvořit sčítáním číslo z argumentu `soucet`.

Každé číslo z pole lze použít libovolněkrát. Všechna čísla jsou nezáporná.

## Jak sečíst (HowSum)

---

Funkce `int[] HowSum(int soucet, int [] cisla)` vrácí pole čísel (vybraných z pole `cisla`) jejichž sečtením lze vytvořit číslo v argumentu `soucet`.

Každé číslo z pole lze použít libovolněkrát. Všechna čísla jsou nezáporná.

## Nejkratší součet (BestSum)

---

Funkce `int[] BestSum(int soucet, int [] cisla)` vrací nejkratší možné pole čísel (vybraných z pole `cisla`), jejichž sečtením lze vytvořit číslo v argumentu `soucet`. Každé číslo z pole lze použít libovolněkrát. Všechna čísla jsou nezáporná.