

Posloupnosti a vektory

E 3011

Jan Böhm

RECETOX

April 5, 2023

Co nás dnes čeká

1 Rozcvička

2 Posloupnosti a řady

3 Vektorová algebra

4 

Fibonacci


Fibonacciho posloupnost je definovaná:

$$F(n) = \begin{cases} 0 & \text{pro } n = 0 \\ 1 & \text{pro } n = 1 \\ F(n-1) + F(n-2) & \text{jinak.} \end{cases}$$

Posloupnost tedy začíná $[0, 1, 1, 2, 3, 5, 8, 13, 21, \dots]$. Vaším úkolem je napsat funkci `Fibonacci(n)`, která vrátí prvních n členů této posloupnosti.

Hint: poslední a předposlední člen posloupnosti `F` získáme pomocí záporného indexu – `F[-1]` a `F[-2]`. Přidávání hodnoty `x` na konec posloupnosti `F` zase zvládne příkaz `F.append(x)`.

Co nás dnes čeká

- 1 Rozcvička
- 2 **Posloupnosti a řady**
- 3 Vektorová algebra
- 4 

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Odhaduje se, že každým rokem se tato populace zvětší o 1.08 %.

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Odhaduje se, že každým rokem se tato populace zvětší o 1.08 %.

V kterém roce můžeme očekávat, že populace na světě přesáhne 10 miliard?

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Odhaduje se, že každým rokem se tato populace zvětší o 1.08 %.

V kterém roce můžeme očekávat, že populace na světě přesáhne 10 miliard?

Malthusovský růst

Jeden z nejstarších modelů růstu populace. Říká, že velikost populace x v následujícím časovém bodě (roce) je přímo úměrná velikosti této populace s koeficientem r – kde r je rozdíl porodnosti a úmrtnosti.

Napište funkci $\text{malthus}(x_0, r, n)$ ve které jsou předvídané velikosti populace v následujících n letech a v současnosti je velikost populace x_0 .

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Kapacita naší planety se odhaduje na 12 miliard lidí. Míra, s jakou populace narůstá, klesá lineárně s velikostí populace - v případě naplněné kapacity prostředí je 0, naopak při prázdném prostředí se odhaduje na 1.62.

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Kapacita naší planety se odhaduje na 12 miliard lidí. Míra, s jakou populace narůstá, klesá lineárně s velikostí populace - v případě naplněné kapacity prostředí je 0, naopak při prázdném prostředí se odhaduje na 1.62.

V kterém roce můžeme očekávat, že populace na světě přesáhne 10 miliard?

Modely růstu populace

Na začátku roku 2023 žilo na zemi 8 miliard lidí. Kapacita naší planety se odhaduje na 12 miliard lidí. Míra, s jakou populace narůstá, klesá lineárně s velikostí populace - v případě naplněné kapacity prostředí je 0, naopak při prázdném prostředí se odhaduje na 1.62.


V kterém roce můžeme očekávat, že populace na světě přesáhne 10 miliard?

Verhulstův model růstu

Tento model říká, že velikost populace x v následujícím časovém bodě (roce) je přímo úměrná velikosti této populace s koeficientem r – kde r je rozdíl porodnosti a úmrtnosti v ideálních podmínkách. Tato porodnost je multiplikativně penalizovaná vyčerpanou kapacitou prostředí - koeficientem z intervalu $(0; 1)$.

Napište funkci $\text{verhulst}(x_0, r, K, n)$ ve které jsou předvídané velikosti populace v následujících n letech a v současnosti je velikost populace x_0 .


Co nás dnes čeká

- 1 Rozcvička
- 2 Posloupnosti a řady
- 3 Vektorová algebra
- 4 

Podívejte se na slajdy o operacích s vektory v přednášce k tomuto týdnu. Měli byste být schopni spočítat všechny operace, které tam jsou zadefinované. Není nutné na všechny tvořit funkce, ale doporučuji si naimplementovat:

- Norma vektoru `norm(u)`.
- Násobení vektoru skalárem `SVP(c, u)` (scalar-vector product)
- Sčítání vektorů `addVectors(u, v)` a zamyslet se, jak vyřešit rozdíl vektorů.
- Skalární součin `dotProduct(u, v)`
- Vektorový součin `crossProduct(u, v)` v trojrozměrném prostoru.
- Smíšený součin `tripleProduct(u, v, w)` v trojrozměrném prostoru.
- Odchylku vektorů `angle(u, v)`.

Co nás dnes čeká

- 1 Rozcvička
- 2 Posloupnosti a řady
- 3 Vektorová algebra
- 4 



Po dnešním cvičení byste měli umět:

- 1 Pracovat s poli. Vytvořit pole, přidávat a odebírat z něj prvky, najít jeho minimum, maximum, součet. Seřadit pole. Zjistit, zda nějaký prvek v něm leží.
- 2 Vytvořit funkci, která na základě nějakého rekurentního vztahu $x(t+1) = f(x(t), x(t-1), \dots)$ generuje členy této posloupnosti.
- 3 Mít implementovány základní funkce pro vektorové počítání.