Numerické metody

E 3011

Jan Böhm

RECETOX

March 27, 2024

Co nás dnes čeká

Babylónská odmocnina

Odvození

Máme číslo x a chceme spočítat $\sqrt{S}$. Jak na to?

1. Nejprve si tipneme. Označíme x_0 jako tip.

Babylónská odmocnina

Odvození

Máme číslo x a chceme spočítat $\sqrt{S}$. Jak na to?

1. Nejprve si tipneme. Označíme x_0 jako tip.
2. Asi jsme se netrefili přesně, označíme ϵ jako chybu. Platí

$$S = (x_0 + \epsilon)^2 = x_0^2 + 2x_0\epsilon + \underbrace{\epsilon^2}_{\approx 0} \approx x_0^2 + 2x_0\epsilon$$

Babylónská odmocnina

Odvození

Máme číslo x a chceme spočítat $\sqrt{S}$. Jak na to?

1. Nejprve si tipneme. Označíme x_0 jako tip.
2. Asi jsme se netrefili přesně, označíme ϵ jako chybu. Platí

$$S = (x_0 + \epsilon)^2 = x_0^2 + 2x_0\epsilon + \underbrace{\epsilon^2}_{\approx 0} \approx x_0^2 + 2x_0\epsilon$$

3. Spočítáme, jak velké chyby ϵ jsme se dopustili $\epsilon = \frac{S - x_0^2}{2x_0}$.

Odvození

Máme číslo x a chceme spočítat $\sqrt{S}$. Jak na to?

1. Nejprve si tipneme. Označíme x_0 jako tip.
2. Asi jsme se netrefili přesně, označíme ϵ jako chybu. Platí

$$S = (x_0 + \epsilon)^2 = x_0^2 + 2x_0\epsilon + \underbrace{\epsilon^2}_{\approx 0} \approx x_0^2 + 2x_0\epsilon$$

3. Spočítáme, jak velké chyby ϵ jsme se dopustili $\epsilon = \frac{S - x_0^2}{2x_0}$.
4. Opravíme náš původní odhad:

$$x_1 = x_0 + \epsilon = x_0 + \frac{S - x_0^2}{2x_0} = \frac{x_0^2 + S}{2x_0}$$

Babylónská odmocnina

Odvození

Máme číslo x a chceme spočítat $\sqrt{S}$. Jak na to?

1. Nejprve si tipneme. Označíme x_0 jako tip.
2. Asi jsme se netrefili přesně, označíme ϵ jako chybu. Platí

$$S = (x_0 + \epsilon)^2 = x_0^2 + 2x_0\epsilon + \underbrace{\epsilon^2}_{\approx 0} \approx x_0^2 + 2x_0\epsilon$$

3. Spočítáme, jak velké chyby ϵ jsme se dopustili $\epsilon = \frac{S - x_0^2}{2x_0}$.
4. Opravíme náš původní odhad:

$$x_1 = x_0 + \epsilon = x_0 + \frac{S - x_0^2}{2x_0} = \frac{x_0^2 + S}{2x_0}$$

5. Vesele iterujeme.

Babylónská odmocnina

Implementace

Implementujte funkci `sqrt(S, tol = 6)`, která spočítá babylónskou odmocninu čísla `S`. Rekurentní formule je:

$$x_n = \begin{cases} \frac{S}{2} & \text{pro } n = 0, \\ \frac{x_{n-1}^2 + S}{2x_{n-1}} & \text{jinak.} \end{cases}$$

Stop kritérium je $|x_n - x_{n-1}| < 10^{-\texttt{tol}}$.

Co nás dnes čeká

Bisekce

Odvození

Máme funkci $f(x)$ a interval $I = (x_L, x_R)$ takový, že f má někde v I *právě jeden* kořen.

Platí, že $sign(f(x_L)) \cdot sign(f(x_R)) \neq 1$. Spočítáme střed tohoto intervalu $m = \frac{x_L + x_R}{2}$ a $f(m)$.

- Pokud $f(m) = 0$ tak jsme našli kořen přesně.
- Pokud $f(m) \neq 0$ a $sign(f(m)) = sign(f(x_L))$, můžeme interval kde kořen je zůžit na (m, x_R).
- Pokud $f(m) \neq 0$ a $sign(f(m)) = sign(f(x_R))$, můžeme interval kde kořen je zůžit na (x_L, m).

Tento postup můžeme opakovat, dokud není interval dostatečně malý. Za přibližné řešení pak můžeme prohlásit například jeho prostřední bod.

Bisekce

Zadání

Implementujte funkci půlení intervalu jako `bisect(fun, l, r, tol = 6)`, která hledá kořen funkce `fun` na intervalu `(l;r)`.
Funkce:

- Zkontroluje, že se liší znaménka funkce `fun` na krajích intervalu `(l;r)`.
- Provede půlení intervalu tolikrát, že výsledný interval má šířku menší, než $10^{-\texttt{tol}}$.
- Vrátí prostředek výsledného intervalu.
- V průběhu vypisuje vhodné informace pro uživatele.

```
def myFun(x):
    return x*sin(x) + 0.5

print(bisect(myFun, 2, 4))
>>> 3.2939743995666504
```

Co nás dnes čeká

Derivace

Derivace

Ne všechny funkce jdou hezky derivovat. Ale snad by šlo aspoň přibližně odhadnout hodnotu derivace funkce v nějakém bodě $f'(x)$.
Definice derivace $f'(x) = \lim_{h\to 0} \frac{f(x+h)-f(x)}{h}$.

`derivative(f, x, tol = 6)`

Napište funkci `derivative(f, x, tol = 6)`, která spočítá přibližnou hodnotu derivace funkce `f` v bodě `x` s přesností $10^{-\texttt{tol}}$.
Využijte definice derivace výše. Začněte s nějakou rozumnou hodnotou h a tuto hodnotu zmenšujte, dokud změna v hodnotě odhadu derivace neklesne pod požadovanou `toleranci`.

Co nás dnes čeká

1. Babylónská odmocnina
2. Půlení intervalu
3. Derivace
4. $\int$
5. 🏠
6. 🍕
7. Test

`integrate(f, a, b, dx = 0.001)`

Napište funkci `integrate(f, a, b, dx = 0.001)`, která počítá přibližně integrál $\int_a^b f(x)\mathrm{d}x$

Vyberte si libovolnou metodu, kterou chcete implementovat – obdélníkovou (levou, pravou, střední, dolní-horní odhad), lichoběžníkovou, nebo klidně nějakou úplně jinou.

Co nás dnes čeká

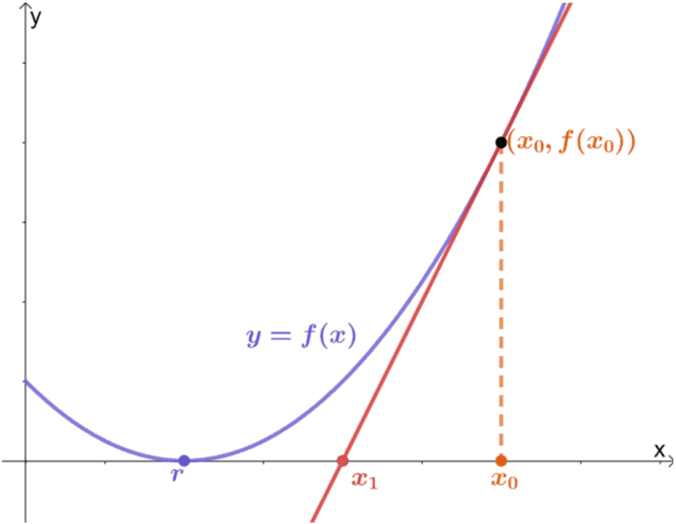
1. Babylónská odmocnina
2. Půlení intervalu
3. Derivace
4. $\int$
5. 🏠
6. 🍕
7. Test

Newtonova metoda

Odvození

Řešíme (přibližně) rovnici $f(x) = 0$, kde f je *hezká* funkce.

1. Začneme tak, že si řešení tipneme: x_0
2. Lokálně nahradíme funkci $f(x)$ přímkou a najdeme místo, kde tato přímka protíná osu x. Toto místo označíme jako x_1.
3. Iterujeme dle chuti.


y
$(x_0, f(x_0))$
$y = f(x)$
x
r
x_1
x_0

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

1. V okolí x_n nahradíme f přímkou (Taylorův polynom 1. stupně, lol).

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

1. V okolí x_n nahradíme f přímkou (Taylorův polynom 1. stupně, lol).
2. Rovnice přímky $(y - y_k) = f'(x_k)(x - x_k)$.

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

1. V okolí x_n nahradíme f přímkou (Taylorův polynom 1. stupně, lol).
2. Rovnice přímky $(y - y_k) = f'(x_k)(x - x_k)$.
3. My chceme takové x_{k+1}, že $y = 0$. Navíc $y_k = f(x_k)$.

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

1. V okolí x_n nahradíme f přímkou (Taylorův polynom 1. stupně, lol).
2. Rovnice přímky $(y - y_k) = f'(x_k)(x - x_k)$.
3. My chceme takové x_{k+1}, že $y = 0$. Navíc $y_k = f(x_k)$.
4. $$-f(x_k) = f'(x_k)(x_{k+1} - x_k) \rightarrow x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Newtonova metoda - odvození

Matematika

Chceme z n-tého odhadu x_n kořene r rovnice $f(x) = 0$ udělat přesnější, $(n+1)$-odhad x_{n+1}.

1. V okolí x_n nahradíme f přímkou (Taylorův polynom 1. stupně, lol).
2. Rovnice přímky $(y - y_k) = f'(x_k)(x - x_k)$.
3. My chceme takové x_{k+1}, že $y = 0$. Navíc $y_k = f(x_k)$.
4. $$-f(x_k) = f'(x_k)(x_{k+1} - x_k) \rightarrow x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Stačí zvolit počáteční odhad x_0 a vymyslet, kdy přestat iterovat.

2. domácí úkol – zadání

Newtonova metoda tečen

Vytvořte funkci `newton(fun, x0, tol = 6)`, která pomocí Newtonovy metody tečen najde (a vrátí) přibližné řešení rovnice `fun = 0` s počátečním odhadem `x0` a s přesností $10^{\texttt{-tol}}$.
Vaše funkce nesmí používat žádné externí knihovny (zejména `math`).
Řešení odevzdejte jako jeden `.py` soubor který obsahuje funkci `newton(fun, x0, tol = 6)` a případné pomocné funkce (např. pro derivaci). Deadline pro odevzdání je neděle 7. 4. 2024 (včetně).

Co nás dnes čeká

1. Babylónská odmocnina
2. Půlení intervalu
3. Derivace
4. $\int$
5. 🏠
6. 🍕
7. Test

Takeaways

Po tomto cvičení byste měli mít připravené funkce

- `sqrt(S, tol = 6)` pro výpočet odmocniny
- `derivative(f, x, tol = 6)`, která spočítá přibližnou hodnotu derivace funkce `f` v bodě `x`
- `bisect(f, x0, y0, tol = 6)` pro numerický výpočet řešení rovnice `f(x) = 0` na intervalu (`x0`;`x0`).
- `integrate(f, a, b, dx = 0.001)`, která numericky integruje funkci `f` na intervalu `(a,b)` s krokem `dx`.

Tyto funkce si přidejte do své "knihovny" `functions.py`.
Po tomto cvičení byste měli umět:

- Implementovat další numerické metody pro řešení nelineárních rovnic jedné proměnné.
- Využívat své funkce pro řešení daných problémů a pokud to problém vyžaduje, funkci upravit.

Co nás dnes čeká

1. Babylónská odmocnina
2. Půlení intervalu
3. Derivace
4. $\int$
5. 🏠
6. 🍕
7. Test

Test

Test

S využitím vašich funkcí s přesností na 6 desetinných míst zodpovězte na následující otázky:

1. Vyčíslete $\sqrt{408849}$
2. Jaký je kořen funkce $\sin(\log^2(x))$ na intervalu $I = \langle 5; 10 \rangle$?
3. Stanovte přibližně derivaci funkce $\sin(\log^2(x))$ v bodě $x = \mathrm{e}$.
4. Spočítejte $\int_0^1 2x^3 \mathrm{e}^{x^2} \mathrm{d}x$.