

Programmering av matematikk i PI tir 14.11 og tir 21.11 Høsten 2023

Du skal programmere matematikk i C++. Bruk gjerne en konsolapplikasjon, det duger til de fleste oppgavene.

Du trenger ikke ta oppgavene i rekkefølge, men de to første er nok de enkleste.

OPPGAVE 1

Lag et program som beregn og skriv ut de 20 første leddene i disse følgene

- a) $\{2n - 1\}_{(n=1)}^{\infty}$ Hva slags følge er dette?
- b) $\left\{(-1)^{n+1} \frac{1}{n}\right\}_{(n=1)}^{\infty}$ Skriv gjerne ut som brøk
- c) $\left\{\frac{1}{n!}\right\}_{(n=1)}^{\infty}$ Skriv gjerne ut som brøk
- d) $\left\{\frac{(-1)^n}{n(n+1)}\right\}_{(n=1)}^{\infty}$ Skriv gjerne ut som brøk
- e) $\left\{\sin\left(\frac{n\pi}{2}\right)\right\}_{(n=1)}^{\infty}$
- f) *Følgen som er definert ved: $F_1 = 1$, $F_2 = 1$, og $F_n = F_{n-1} + F_{n-2}$ for $n \geq 3$*
Hva slags følge er dette? Hva kalles et når følgen er angitt på denne måten?

OPPGAVE 2 Sigma

Lag et program som skriver ut alle leddene OG summen:

- a) $\sum_{i=1}^{20} (i^2 - 4)$
- b) $\sum_{i=0}^{32} \sin\left(\frac{i}{10}\right)$
- c) $\sum_{i=0}^{16} \sqrt{4i}$

OPPGAVE 3

Lag ei klasse som kan beregne funksjonsverdi for en del funksjoner. Kall klassa mittLilleMattebibliotek (eller noe annet)

Du skal programmere kjente funksjoner ved å bruke potensrekka for disse funksjonene, med sentrum i 0, dvs. $a=0$. Da kalles det Maclaurin rekka for de aktuelle funksjonene, disse står på side 351 i boka.

- Lag en funksjon for $\sin(x)$ med input parametre n og x , som returnerer S_n og $|E_n|$.
- Lag en funksjon for $\cos(x)$ med input parametre n og x , som returnerer S_n og $|E_n|$.
- Lag en funksjon for e^x med input parametre n og x , som returnerer S_n og $|E_n|$.
- Lag en funksjon som beregner $\sin(x) + \cos(x)$ med input parametre n og x , som returnerer S_n og $|E_n|$.

TIPS: Tenk på oppgave c før du programmerer a og b, slik at designet passer for c også.

- Ekstraoppgave: Potensrekka for en funksjon er mest nøyaktig rundt sentrum for rekka, som her er $x=0$. For sinus og cosinus er det egentlig nok å beregne verdien i første kvadrant, og deretter bruke det vi vet om at sinus og cosinus er periodiske, samt hvilke kvadrant de er positive og negative i. Bruk denne kunnskapen til å forbedre din beregning av $\sin x$ og $\cos x$ i a og b slik at de blir ganske nøyaktige for alle verdier av x uten at du har alt for stor n . NB! Isteden for å bruke potensrekka til å beregne $\sin x$ og $\cos x$ i første kvadrant, kan en beregne for $-\frac{\pi}{4} \leq x \leq \frac{\pi}{4}$

OPPGAVE 4

Bruk av mittLilleMattebibliotek

- Kall din funksjon for $\sin x$ og skriv ut: S_n , $|E_n|$, S og $S-S_n$ (der S er $\sin x$ beregnet med C++ sitt mattembibliotek) for $n=5$ hver $\pi/3$ fra $x = -5\pi/3$ til $x=5\pi/3$

Legg merke til hvor din S_5 er en god tilnærming og hvor det er store avvik.

- Kall din funksjon for $\cos x$ og skriv ut: S_n , $|E_n|$, S og $S-S_n$ (der S er $\cos x$ beregnet med C++ sitt mattembibliotek) for $n=5$ hver $\pi/3$ fra $x = -5\pi/3$ til $x=5\pi/3$

Legg merke til hvor din S_5 er en god tilnærming og hvor det er store avvik.

- Kall din funksjon for e^x og skriv ut: S_n , $|E_n|$, S og $S-S_n$ (der S er e^x beregnet med C++ sitt mattembibliotek) for $n=5$ hver 0,5 fra $x=-4$ til $x=4$

Legg merke til hvor din S_5 er en god tilnærming og hvor det er store avvik.

- Kall din funksjon for $\sin(x) + \cos x$ og skriv ut: S_n , $|E_n|$, S og $S-S_n$ (der S er $\sin x + \cos x$ beregnet med C++ sitt mattembibliotek) for $n=5$ hver $\pi/3$ fra $x = -5\pi/3$ til $x=5\pi/3$

Legg merke til hvor din S_5 er en god tilnærming og hvor det er store avvik.

- Ekstraoppgave (ikke nødvendig for forståelsen, men gjør det mer visuelt): Tegn opp kurver med S og S_n for oppavene ovenfor. Du trenger da et grafikkbibliotek for C++, f.eks ImGui og ImPlot (disse hører sammen)

OPPGAVE 5 Numerisk integrasjon; Riemann-sum

Eks. 7.1.4 i boka

Gitt funksjonen $f(x) = x^3 - 3x^2 + 2x + 1$

Vi skal beregne $\int_0^{2.5} f(x)dx$ numerisk, med midtpunktsmetoden. Vi deler intervallet fra 0 til 2.5 i tusen like store deler, og får da formelen:

$$R_{1000} = \sum_{i=1}^{1000} f(x_i) \cdot \Delta x$$

$$\text{der } \Delta x = \frac{2.5}{1000}$$

$$\text{og } x_i = (i - 0.5) \cdot \Delta x$$

- Lag en funksjon som beregner og skriver ut R_{1000}
- Lag funksjonen slik at $a = x_{\text{fra}}$ og $b = x_{\text{til}}$ og antall intervaller n er parametre til funksjonen (det kan deles opp i flere funksjoner)
- Programmer f'' i en funksjon også, og beregn maksimal størrelse på feilen ut fra a, b og n

7.7.2 SETNING

Forutsatt at den dobbeltderivate $f''(x)$ er kontinuerlig på intervallet $[a, b]$ og vi har valgt en konstant M_2 som oppfyller $M_2 \geq |f''(x)|$ for $x \in [a, b]$, gjelder følgende ulikhet for feilen ved midtpunktregelen:

$$|E_n| \leq \frac{M_2(b-a)^3}{24n^2}$$

- Gjør programmet generelt, slik at det enkelt kan brukes til andre funksjoner
- Beregn det ubestemte integralet $F(x) = \int x^3 - 3x^2 + 2x + 1$, og lag en funksjon som beregner denne for en gitt x -verdi. Skriv ut x , $F(x)$ og Riemann sum så langt for denne x for hver 0.5 x -verdi, dvs. for $x=0$, $x=0.5$, $x=1$ osv.

OPPGAVE 6

Programmer eksempel 7.7.11 og 7.7.13, dvs. Simpsons regel med feilestimering.

OPPGAVE 7 Utfordring

Lag enhetstester for funksjonene.

Dersom en lager enhetstestene på forhånd, kalles det Test-driven development (TDD). Se

https://en.wikipedia.org/wiki/Test-driven_development

Bruken er omtrent som følger: Du lager testene på forhånd, som en del av designet. Pass på også å lage tester for grensetilfellene, f.eks. minimumsverdi, maksimumsverdi for input-parametrene. Første gang du kjører testene skal alle testene feile. Deretter programmerer du funksjonen slik at alle testene går igjennom.