INSTRUKSIES:

(a) Huiswerke moet met die aanvang van die tutoriaalperiode op Vrydae ingehandig word. Geen elektroniese huiswerke word aanvaar nie en geen huiswerke sal laat ingeneem word nie. In die praktyk beteken dit dat jy moet mik om teen Donderdagaand klaar te wees om toe te laat vir moontlike probleme met drukkers, skerms wat vries, en so aan.

(b) Samewerking op hierdie huiswerke word beperk tot die uitruil van enkele idees en wenke. Die uitruil van data, grafieke, rekenaarprogramme of die besonderhede van wiskundige berekenings is nie toelaatbaar nie. Wat jy inhandig moet jou eie werk wees.

Probleem 1: 'n Belangrike probleem in die numeriese wiskunde is die effektiewe evaluering van polinome

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n.$$

(a) Definieer die hoeveelhede p_0, p_1, ..., p_n soos volg

$$p(x) = \underbrace{\underbrace{\underbrace{a_0}_{p_0} + a_1 x}_{p1} + a_2 x^2 + \ldots + a_n x^n}_{p_n}.$$

Die *direkte* of *voorwaartse rekursie* vir die evaluering van $p(x)$ vir gegewe x kan dan gedefinieer word deur

$$q_k := x\, q_{k-1}, \qquad p_k := p_{k-1} + a_k\, q_k, \qquad k = 1, 2, \ldots, n,$$

met beginwaardes

$$p_0 := a_0, \qquad q_0 := 1.$$

Tel die totale aantal optellings en vermenigvuldigings wat hierdie algoritme uitvoer.

(b) Beter is **Horner** se algoritme, gebaseer op

$$p(x) = \underbrace{a_0 + x(a_1 + x(a_2 + \ldots \underbrace{(a_{n-1} + x \underbrace{a_n}_{u_n})}_{u_{n-1}} \ldots))}_{u_0}.$$

Dit kan as 'n *terugwaartse rekursie* geïmplementeer word

$$u_k := a_k + x\, u_{k+1}, \qquad k = n-1, n-2, \ldots, 0,$$

met beginwaarde $u_n := a_n$, en endwaarde $u_0 := p(x)$. Tel die totale aantal optellings en vermenigvuldigings wat hierdie algoritme uitvoer, en vergelyk met die algoritme van deel (a).

(c) Implementeer Horner se algoritme in 'n MATLAB programmetjie, met die roep-instruksie `p = horner(a,x)`. Die vektor `a` bevat die koëffisiënte $a_0, a_1, \ldots, a_n$, en `x` is die waarde van x waar $p(x)$ bereken moet word. Toets jou program deur die afvoer met MATLAB se ingeboude funksie `polyval` te vergelyk.[1]

Probleem 2: Beskou die Taylor/Maclaurin polinoombenadering tot die eksponensaalfunksie

$$p(x) = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \ldots + \frac{1}{n!}x^n, \qquad p(x) \approx e^x.$$

(a) Gebruik $p(x)$ om e^x te benader by $x = -1$ vir waardes $n = 1, 2, \ldots 20$. Stip die *relatiewe fout* as funksie van n (gebruik 'n logskaal op die vertikale as met `semilogy`). Word volle akkuraatheid bereik?

(b) Herhaal deel (a) met $x = -10$, en laat $n = 1, 2, \ldots 80$. Word volle akkuraatheid bereik? Indien nie, stel vas waar die verlies aan akkuraatheid plaasgevind het.

(c) Verander die algoritme van deel (b) deur e^x te bereken vir $x = +10$, en gebruik dan $e^{-10} = 1/e^{+10}$. Is volle akkuraatheid nou bereik?

Probleem 3: Beskou die integrale

$$y_n = \int_0^1 t^n e^{-t}\, dt, \qquad n = 0, 1, 2, \ldots$$

(a) Toon aan dat vir alle $n = 0, 1, 2, \ldots$

$$\frac{1}{e(1+n)} < y_n < \frac{1}{(1+n)},$$

en lei af dat $y_n \to 0$ as $n \to \infty$.

(b) Deur deelwyse integrasie, toon aan dat

$$y_n = n y_{n-1} - 1/e, \qquad n = 1, 2, \ldots \tag{1}$$

met

$$y_0 = 1 - 1/e.$$

(c) Implementeer die rekursie van deel (b) om y_n te bereken vir $n = 0, 1, 2, \ldots, 25$. Is die numeriese resultate in ooreenstemming met die teoretiese resultate van deel (a)?

(d) Verklaar die onstabiliteit teoreties, deur die effek van 'n klein afrondingsfout, ϵ, in die aanvangswaarde te modelleer.

(e) Herskryf die rekursie (1) in "terugwaartse" vorm, d.w.s., sodat y_{n-1} in terme van y_n bereken word. Gebruik 'n beginwaarde $y_m = 0$, met $m = 40$ om y_n te bereken vir $n = 25, 24, 2, \ldots, 1, 0$. Bereken ook y_0, y_5, y_{10}, y_{15} en y_{20} tot sestiensyfer akkuraatheid met die numeriese integrasie program van Mathematica (of enige ander pakket). Lewer die terugwaartse rekursie volle akkuraatheid?

[1] Tik `help polyval` in MATLAB om meer te leer.