TFY4115 Fysikk (MTELSYS/MTTK/MTNANO)
Tips for ekstraøving

Oppgave 1.

b. Tegn inn alle kreftene som virker på mannen i posisjon B. Sett opp Newtons 2. lov for kreftenes komponent i radiell retning (sentripetalakselerasjon). Hastigheten v har du funnet i **a.** Brettkjøreren vil lette fra underlaget når $F_{\mathrm{N}}(\theta_0) = 0$.

c. Du må velge Matlab-variable (vektorer) for de aktuelle fysiske størrelser $\alpha, \theta, \omega, F_{\mathrm{N}}$ og t. Foreslår du velger henholdsvis `alfa, theta, omega, FN` og `t`. Og du må velge verdi for tidsinkrement Δt, f.eks. `dt=0.01`. Likningene

$$\alpha = \frac{a}{R} = \frac{g \sin\theta}{R} \quad , \qquad \mathrm{d}\omega = \alpha \cdot \mathrm{d}t \quad , \qquad \mathrm{d}\theta = \omega \cdot \mathrm{d}t \tag{1}$$

gir hvor mye ω og θ øker for hvert tidsinkrement `dt`, på grunnlag av vinkelakselerasjonen α fra forrige iterasjon. Hver iterasjon kan f.eks. skrives slik:

```
omega(n+1) = omega(n) + alfa(n)*dt;
theta(n+1) = theta(n) + omega(n)*dt;
alfa(n+1) = g*sin(theta(n+1))/R;
```

Det kan være gunstig å bruke en WHILE-løkke til iterasjonene isf. FOR-løkke, fordi iterasjonen skal avbrytes når normalkrafta er blitt null.

Før du går inn i WHILE-løkka må du initialisere alle verdier ved `t(1)=0`. Hvis du starter iterasjonen på toppen med $\omega = 0$ og $\alpha = 0$ vil vil brettkjøreren ikke bevege seg. WHILE-løkka vil da bli uendelig. Du må derfor legge inn en liten ustøhet, enten ved å starte med en liten vinkel f.eks. `theta(1)=0.01` eller med en liten starthastighet `omega(1)`> 0. Anbefaler det første. Du bør i tillegg teste WHILE-løkka mot et stort antall max iterasjoner for å unngå uendelig løkke pga. programmeringsfeil.

Tallverdi for alle størrelser angir du i standard SI-enheter, dvs. rad, rad/s, s osv. Siden vi regner på akselerasjon og ikke krefter, er det naturlig isf. F_{N} i newton å bruke F_{N}/m. Bevegelsen er uansett uavhengig av massen m.

Bevegelseslikningene gjelder ikke når brettkjøreren har lettet fra underlaget (se punktet ovenfor), så du bør stoppe simuleringen når $F_{\mathrm{N}} < 0$. (I alle tilfelle må du stoppe ved `theta`$=\pi/2$.) Du må derfor i løkka også regne ut FN, og selvfølgelig øke tid og telleparameter for hver gjennomgang av løkka:

```
FN(n+1) = g*cos(theta(n+1)) - omega(n+1)*omega(n+1)*R;
t(n+1) = t(n)+dt;
n = n+1;
```

Når løkka er fullført, plott resultatet som angitt i oppgaven. Forslag til plotting av to grafer ved siden av hverandre og med høvelig tekst, finner du i matlab-scriptet utlagt på øvingens nettside. Her finner du faktisk et fullstendig Matlabprogram for oppgaven (uten friksjon).

Rapportering av enkelte tallverdier til skjermen kan gjøres f.eks. ved

```
fprintf(1,'FN=0 når t=%5.2f theta=%5.2f v=%5.2f \n', t(n), theta(n)*180/pi, omega(n)*R);
```

En alternativ numerisk algoritme er Verlet-integrasjon. Den ikke-lineære pendelen i Øving 5 med null demping har akkurat samme bevegelseslikninger som rullebrettkjøreren. Du kan derfor kopiere algoritmen herfra. I tillegg må du beregne ω og α for hver iterasjon, for å kunne beregne F_{N}.

Oppgave 2.

a. Med friksjon μ blir det bare små modifikasjoner i likn. (1), idet akselerasjonen blir mindre:

$$\alpha = \frac{a}{R} = \frac{g \sin\theta - \mu F_{\mathrm{N}}}{R} \tag{2}$$

I WHILE-løkka må altså `alfa` beregnes etter `FN` er beregnet. Iterasjonene må starte først ved en vinkel hvor tyngden overvinner friksjonen: $mg \sin\theta > \mu mg \cos\theta$, dvs. $\tan\theta > \mu$. Det betyr at i programmet må θ initialiseres til f.eks. `theta(1) = atan(my)+0.01`.

b. Nå må friksjonsarbeidet inngå i energibevarelsen. Sett opp energibalansen på differensiell form:

$$\mathrm{d}(mgh) + \mathrm{d}\left(\frac{1}{2}mv^2\right) \quad = \quad \mathrm{d}W_{\mathrm{f}}$$

der $h = R\cos\theta$ og $\mathrm{d}W_{\mathrm{f}} = \vec{F}_{\mathrm{f}} \cdot \mathrm{d}\vec{s} = -\mu_{\mathrm{k}} F_{\mathrm{N}} \cdot R\,\mathrm{d}\theta$ med uttrykk for F_{N} fra oppgave 1.

Et krav i likningen er at en positiv $\mathrm{d}\theta$ nødvendigvis må gi en økning i fart, dvs. positiv $\mathrm{d}v$.