Noen programmeringsoppgaver i R

28. april 2015

1 Oppgaver

1.1 Funksjoner

Lag en funksjon som beregner arealet av en sirkel med en gitt radius r og bruk funksjonen til å beregne areal til sirkler med radius 1, 2, 10 og 20.

1.2 Renteberegning

Lag en funksjon som beregner størrelsen på en bankinnskudd x etter t år gitt at årlig rente er r %.

1.3 Funksjoner som kaller en annen funksjon gitt som argument

Et integral kan beregnes numerisk med (den naive) tilnærmingen

$$\int_a^b f(x)dx = \frac{b-a}{n} \sum_{i=1}^{n} f\left(a + (b-a)\frac{i-1/2}{n}\right). \tag{1}$$

For $a = 10$, $b = 15$ og $n = 10$ er tilnærmingen m.a.o. gitt ved

$$0.5(f(10.25) + f(10.75) + f(11.25) + \ldots f(14.75)). \tag{2}$$

Programmer en funksjon som beregner et slikt integral av en vilkårlig funkjon f fra a til b numerisk ved hjelp av tilnærmingen over. La f, a, b og n være innargument til funksjonen. Du må altså i tillegg programmere en funksjon som representerer f. Test at du får tilnærmet rett svar ved å beregne et integral med kjent løsning. Sammenlign også med resultatet du får om du bruker `integrate`-funksjonen i R.

1.4 If-setning

1.5 Repeat løkke

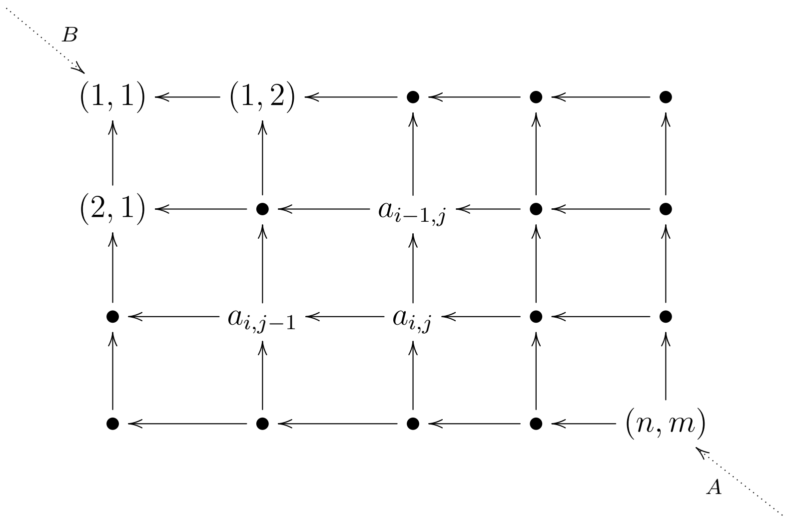
1.6 Lett oppgave..

1.7 Debugging

Følgende funksjon skal... Finn ut hva som er feil, f.eks. ved bruk av debug() i Rstudio.

1.8 Nøstet for-løkke og matriser

Anta at vi står i punkt A (kafe de ni muser) i rutenettet vist i figur 1 og at vi skal velge en kortest mulig vei (langs pilene) til punkt B (krambua). Programmer en funksjon som beregner antall veier fra A til B for vilkårlig valg av m og n.



Figur 1: Kart over midtbyen...

Tips: La $a_{i,j}$ betegne antall mulige veivalg til punkt B fra kryss (i, j) i rutenettet. I fra kryss langs øvre og venstre kant i rutenettet vil det åpenbart være bare en vei til B ($a_{i,j} = 1$ for $i = 1$ og $j = 1$). Anta i tillegg at vi har greid å beregne $a_{i-1,j}$ og $a_{i,j-1}$ (se figur). Hva kan vi da si om antall veier fra kryss (i, j)? Bruk svaret på dette spørsmålet når du skal programmere løsningen av oppgaven.

Hint: Løsningen kan finnes iterativt ved bruk av en nøstet (dobbel") for-løkke samt matriser.

1.9 Rekursive funksjoner (litt vanskelig)

Rekursive funksjon kaller seg selv under visse forutsentinger. Ved å utnytte at $n! = n(n-1)!$ kan for eksempel $n! = 1 \cdot 2 \cdot \ldots \cdot n$ beregnes med den rekursive funksjonen

```
minfakultet <- function(n) {
  if (n == 1)
    1
  else
    n*minfakultet(n-1)
}
```

Løs problem 1.8 ved i stedet å programmere en rekursiv funksjon.

1.10 Rekursiv tegning av et tre

Programmer en funksjon som i et eksisterende tomt plot tegner en strek av lengde r i retning gitt ved en vinkel θ (målt i radianer) i fra punktet x, y til punktet $x + r\cos(\theta), y + r\sin(\theta)$. Størrelsene x, y, r og θ må med andre ord være innargument. Gitt at r er større enn f.eks. 0.1 skal funksjonen gjøre to kall til seg selv slik at den tegner to mindre streker av lengde $0.8r$ i fra punktet $x + r\cos(\theta), y + r\sin(\theta)$ i retninger $\theta + \pi/4$ og $\theta - \pi/4$. La eventuellt også endring i retning, med hvilken faktor grenene endrer lengde, og minste grenlengde også være innargument og plot ulike trær gitt ulike valg av disse parameterne.

Et tomt plot kan lages med uttrykket

```
plot(NA, xlim = c(-2,2), ylim = c(-2,2))
```

1.11 Primtall

Lag en funksjon som beregner hvilke av de første n heltallene som er primtall og ikke primtall ved å eliminere tall som ikke er primtall, m.a.o. alle tall som er multiplum av 2,3,4,5,6, d.v.s. først tallene 2,4,6,8,..., så tallene 3, 6, 9, 12, ..., o.s.v. Dette kan gjøre ved hjelp av en dobbel løkke og en logisk vektor. Flere løsninger er mulige.

1.12 Lag din egen oppgave

Lag din egen oppgave ved å editere dette dokumentet som er å finne på `https://www.sharelatex.com/project/552537b13350a5f31e2362c3`

2 Løsninger