

## Introduktion

Läs igenom avsnitt 1-5 av Jörgen Löfströms MATLAB-handledning och lös tillhörande övningsuppgifter. Handledningen hittar du via en länk på kurshemsidan.

## Logik

I MATLAB finns de logiska operationerna AND, OR och NOT, som också kan skrivas med hjälp av tecknena  $\&$ ,  $|$  och  $\sim$ . Använd HELP för att ta reda på hur de fungerar. Lös nedanstående uppgifter genom att använda dessa operationer. Man behöver också använda kommandona FOR och LENGTH. Ett generellt tips är att vid lösning av uppgifterna 2-3 har man alltid nytta av att använda funktionen man skapade i steget innan.

1. **Majoritetsoperatorn:** Skapa en funktion av tre signaler (0:or eller 1:or) vars värde överensstämmer med majoriteten av de tre insignalerna.
2. **Heladderaren:** Skapa en funktion som adderar tre ensiffriga binära tal.
3. **Adderaren:** Skapa en funktion som adderar två godtyckliga tal på binär form. (De två binnära talen, liksom svaret, anges lämpligen som radvektorer bestående av nollor och ettor.)

## Heltalsaritmetik

1. MATLABs funktion "mod" fungerar dåligt för beräkning av storheter av typen

$$m^n \text{ mod } r$$

så fort  $m$ ,  $n$  och  $r$  är någotsånär stora, pga att modulatoräkningen inte utförs förrän  $m^n$  är uträknat. Därför blir din uppgift att skapa en funktion som beräknar  $m^n \text{ mod } r$  på ett sätt som gör att man kan räkna med så stora tal som möjligt. Eftersom  $m^4 = (m^2)^2$ ,  $m^8 = (m^4)^2$  etc är det lämpligt att utveckla  $n$  binärt. Därför är det lämpligt att bärja med att skapa en funktion som från ett heltal skapar en radvektor som innehåller dess binärutveckling. Man bör också skapa en funktion som kan multiplicera två stora heltal utan att "tappa siffror". Detta kan till exempel göras med följande funktion, kallad "prodmod":

```
function y=prodmod(a,b,n)
a=mod(a,n);
b=mod(b,n);
```

```
if a>b,
```

```
c=b;  
b=a;  
a=c;  
end
```

```
u=tenary(a);  
s=length(u);  
y=0;  
c=b;
```

```
for i=s:-1:1,  
y=mod(y+c*u(i),n);  
c=mod(10*c,n);  
end
```

Funktionen “tenary” som dyker upp här ser ut så här:

```
function y=tenary(x)  
  
s=max(0,floor(log10(x+0.5)));  
y=ones(1,s);  
  
rest=x;  
  
for i=s+1:-1:1,  
y(i)=floor(rest/10^(i-1));  
rest = mod(rest,10^(i-1));  
end  
y=y(s+1:-1:1);
```

och gör om ett heltal till en vektor som består av siffrorna i talet. En funktion som skapar en binärutveckling fås genom att byta ut 10 mot 2 i “tenary”.

2. Skapa ett program som läser och skickar RSA-ktypterad kod. ( $A=1$ ,  $B=2$ , ...,  $\ddot{O}=28$ , mellanslag=29). Du behöver en nyckel och därmed två stora primtal. I MATLAB finns funktionen “isprime” som testar om ett tal är ett primtal för tal upp till ca 10 siffror, så använd inte större tal än så. Skicka och ta emot meddelanden till och från andra grupper.

## Grafer och sannolikheter

1. Skapa en kolonn med 100 slumpstal som är
  - (a) likformigt fördelade över  $(0, 1)$ ,
  - (b) likformigt fördelade över  $\{1, 2, 3, 4, 5, 6\}$ ,
  - (c) binomialfördelade med parametrar 20 och  $1/3$ .

**Tips:** Använd kommandot RANDOM.

2. (a) Låt  $G = G(V, E)$  vara grafen där

$$V = \{a, b, c, d, e, f, g, h, i, j, k\}$$

$$E = \{ab, ac, ak, bc, bd, be, cd, de, df, dg, eh, ei, fi, fj, gj, gk, ik\}.$$

(Obs. att beteckningarna för kanterna är på kortform. Exempelvis betyder  $ak$  egentligen  $\{a, k\}$ .) Låt  $X_0, X_1, X_2, \dots$  vara en slumpvandring på  $G$ . **Simulera** med datorns hjälp fram den approximativa fördelningen för  $X_{10}$  och  $X_{30}$ .

Skriv också ett program som **beräknar** den exakta fördelningen för  $X_{10}$  och  $X_{30}$ .

- (b) Gör samma saker som i (a) för slumpvandring på den **riktade** grafen  $G = G(V, E)$  där

$$V = \{a, b, c, d, e, f, g\}$$

$$E = \{ab, ac, ag, bc, bd, ca, cd, df, eb, ed, ee, fc, ff, fg, ge\}.$$

(Obs. att här betyder exempelvis  $ag$  kanten **från**  $a$  **till**  $g$ , till skillnad från i (a) där riktningar saknas.)

I uppgift (a), vad är den stationära fördelningen?