

Block 2

Algebra och Diskret Matematik A

BLOCK INNEHÅLL

[Referenser](#)

[Inledning](#)

1. [Följder](#)
2. [Rekursiva definitioner](#)
3. [Sigmanotation för summor](#)
4. [Strängar](#)
5. [Tal](#)
6. [Övningsuppgifter](#)

Följder, strängar och tal

Referenser

[J] Edition 5: Nedanstående text + [J] 2.2, 2.3, 5.1

[J] Edition 4: Nedanstående text + [J] (2.2), (2.3), (5.1)

[J] Edition 6: Nedanstående text + [J] [2.3], [5.2], [7.1]

Bemärk, referenser i texten utan parenteser är till [J]ohnsonbaugh Edition 5, referenser i ()-parenteser är till [J] Edition 4, och referenser i []-parenteser är till [J] Edition 6.

Inledning

I detta block skall vi först lära oss den grundläggande terminologin för ändliga, oändliga talföljder och strängar. Därefter skall vi se på, hur vårt talsystem är uppbyggt, och vi introducerar det binära och det hexadecimala talsystemen, som båda används inom t.ex. datalogi. Vi skall också lära oss att räkna i dessa två nya talsystem, och vi kommer att utveckla algoritmer för att konvertera tal från ett talsystem till ett annat.

1. Följder

Börja med att läsa avsnitt 2.2 (2.2) [2.3] i [J] från överst på sida 64 (73) [nederst sid 103] fram till definition 2.2.12 sida 66 (76) [2.3.14 sida 107]. Här definieras **följder** och **delföljder**.

En oändlig/ändlig följd är en oändlig/ändlig **ordnad** lista av symboler (som regel tal), där varje element i listan har ett nummer, som vi kallar elementets **index**. En generell oändlig följd har inget sista element och ser därför ut som:

$$s_1, s_2, s_3, s_4, s_5, \dots,$$

där varje s_n , $n=1, 2, 3, \dots$, är en symbol. En ändlig följd har ett sista element och är därför på formen

$$s_1, s_2, s_3, s_4, s_5, \dots, s_t,$$

där 1 är följdens första index och t är följdens sista index.

Observera, att en oändlig följd

$$s_1, s_2, s_3, s_4, s_5, \dots$$

kan anges på flera olika sätt. I [J] används följande tre:

1. "låt s vara en följd";
2. "låt $\{s_n\}$ vara en följd";
3. "låt $\{s_n\}_{n=1}^{\infty}$ vara en följd".

Notationerna under punkt 2. och 3. är en aning osedvanliga. Det är mer vanligt att skriva (s_n) och $(s_n)_{n=1}^{\infty}$ i stället för $\{s_n\}$ och $\{s_n\}_{n=1}^{\infty}$. Grunden till detta är, att parenteserna $\{$ och $\}$ används för att markera mängder, som vi såg i block 1, och följder är *inte* mängder. Mängder är *oordnade* samlingar av objekt, medan en av de mest karakteristiska egenskaperna hos en följd är, att den är *ordnad*.

Övre indexet ∞ i $\{s_n\}_{n=1}^{\infty}$ utläses som "oändlig" och används för att indikera, att följden inte har något sista index och därför inte något sista element. Undre indexet $n=1$ i $\{s_n\}_{n=1}^{\infty}$ används för att ange att följdens första index är 1, och följdens första element är därför s_1 . Om följden är ändlig och har s_t som sista element, anger man i övre indexet följdens sista index, t , i stället för ∞ . På samma sätt är det ibland klokt att låta en följdens första index vara ett annat nummer än 1, t.ex. 0, och i sådana fall anger man detta genom att låta undre indexet vara " $n=0$ " i stället för " $n=1$ ", d.v.s.

$$\{s_n\}_{n=0}^{\infty}$$

Det påstås efter exempel 2.2.11 (2.2.11) [2.3.13] i [J], att man kan konstruera nya följder genom att addera och multiplicera följders element. Vi ger ett exempel på detta:

Exempel

Låt u vara följden

$$2, 4, 6, 8, 10, \dots,$$

dvs. $u_n = 2n$, for $n=1, 2, 3, \dots$. Låt vidare v vara följden

$$1, 3, 5, 7, 9, \dots,$$

dvs. $v_n = 2n - 1$, for $n=1, 2, 3, \dots$. Vi kan då definiera en följd s genom att låta $s_n = u_n + v_n$ för $n=1, 2, 3, \dots$, dvs. s är följden

$$2+1, 4+3, 6+5, 8+7, 10+9, 12+11, \dots,$$

som är lika med följden

$$3, 7, 11, 15, 19, 23, \dots,$$

dvs. $s_n = 4n-1$ for $n=1, 2, 3, \dots$, som det också kan ses av att

$$s_n = u_n + v_n = (2n) + (2n-1) = 4n - 1.$$

På samma sätt kan vi definiera följden $p_n = u_n \cdot v_n$ för $n=1, 2, 3, \dots$, dvs. p är följden

$$2 \cdot 1, 4 \cdot 3, 6 \cdot 5, 8 \cdot 7, 10 \cdot 9, 12 \cdot 11, \dots$$

som är lika med följden

$$2, 12, 30, 56, 90, 132, \dots$$

Denna följdens n :te element kan beräknas genom

$$p_n = u_n \cdot v_n = (2n) \cdot (2n-1) = 4n^2 - 2n,$$

vilket skulle ha varit svårt att gissa om vi inte på förhand visste att $p_n = u_n \cdot v_n$. $\square$

2. Rekursiva definitioner

Läs nu avsnitt 5.1 (5.1) [7.1] i [J] från sidan 224 (256) [279] till och med exempel 5.1.8 (5.1.8) [7.1.8] sidan 228 (261) [283-284], men inte beviset på sidan 229 (262) [284]. Vad du bör kunna göra när du är klar med avsnitt [J] 5.1 (5.1) [7.1] är att bestämma en rekursionsformel från några regler som i exempel 5.1.3 (5.1.3) [7.1.3] och 5.1.6 (5.1.6) [7.1.6].

På sida 224-225 (256-257) [279-280] hittar vi definitionen på en **rekursionsformel**. Observera att för att bestämma en följd, är det inte tillräckligt med bara en rekursionsformel. Man behöver också **startvillkoren**.

[J] Exempel 5.1.2 (5.1.2) [7.1.2]:

I exempel 5.1.2 (5.1.2) [7.1.2] ges definitionen av Fibonacciföljden. Observera att det finns olika sätt att definiera denna.

[J] Exempel 5.1.3 (5.1.3) [7.1.3]:

'12% compounded annually' betyder '12% ränta på årsbasis', dvs. att man lägger 12% ränta till kapitalet en gång varje år. Om man t.ex. börjar med

$$A_1 = \$1000$$

den 1 januari 2000, så är det fortfarande \$1000 den 31 december 2000, men den 1 januari 2001 ökas detta till

$$A_2 = \$1120 (\$1000 + \text{ränta, som är } \$1000 \cdot 12/100 = \$120).$$

Vi ser att

$$1120 = 1000 + 1000 \cdot 12/100 = 1000(1 + 12/100) = 1000(1,12).$$

Om vi sätter $A_1 = 1000$, så har vi

$$A_2 = 1,12A_1.$$

Det är precis samma sak om vi börjar med år $n-1$ och kapital A_{n-1} . Nästa år har vi har vi

$$A_n = 1,12A_{n-1}.$$

I boken ser vi, med hjälp av iterationen, att

$$A_n = (1,12)^n A_1 = (1,12)^n 1000.$$

[J] Exempel 5.1.5 (5.1.5) [7.1.5]:

Här är S_n antal delmängder i en mängd med n elementer. Vi har studerat mängden av alla delmängder i en mängd X av storlek n i [J] 2.1, Theorem 2.1.4 (2.1.4) [2.1.6], och [J] 4.1 (4.1) [6.1], Example 4.1.5 (4.1.4) [6.1.5]. Kom ihåg att mängden av alla delmängder i en mängd X kallas för **potensmängden** $\mathcal{P}(X)$ till X .

I boken bestäms att $S_n = 2S_{n-1}$ och $S_0 = 1$. Med hjälp av rekursionsformeln får vi

$$S_n = 2S_{n-1} = 2^2S_{n-2} = 2^3S_{n-3} = 2^4S_{n-4} = \dots = 2^{n-1}S_1 = 2^nS_0 = 2^n.$$

[J] Exempel 5.1.6 (5.1.6) [7.1.6]:

Varje sträng av längd n börjar med ett 0 eller 1, som efterföljs av $n-1$ andra tal.

FALL 1	0	...(n-1 bitar).....
FALL 2	1	...(n-1 bitar).....

Vi vill inte ha 111 någonstans. Om de n-1 bitarna som följer efter 0 i fall 1 inte innehåller någon 111, då innehåller hela strängen ingen 111. Är det samma sant om de n-1 bitarna som följer efter 1 i fall 2 inte innehåller någon 111? Nej, om de n-1 bitarna börjar med en nolla, och inte innehåller någon 111 går det bra, men om de börjar med en etta, vad gäller då?

FALL 1	0	...(n-1 bitar).....	
FALL 2a	1	0	...(n-2 bitar).....
FALL 2b	1	1	...(n-2 bitar).....

I fall 2b, om första av de n-2 bitarna är en nolla går det bra men om det är en etta går det inte bra (Vi får 111!). Så de enda tre fallen som kan vara utan 111 är

FALL 1	0	...(n-1 bitar).....		
FALL 2a	1	0	...(n-2 bitar).....	
FALL 2b	1	1	0	...(n-3 bitar).....

Antalet i varje fall är S_{n-1} , S_{n-2} , respektive S_{n-3} , så totalt är S_n lika med $S_{n-1} + S_{n-2} + S_{n-3}$, dvs.

$$S_n = S_{n-1} + S_{n-2} + S_{n-3}.$$

Läs nu exempel 5.1.7-5.1.8 (5.1.7-5.1.8) [7.1.7-7.1.8] i [J].

3. Sigmanotation för summor

Det är ofta nödvändigt att skriva summan av alla element från en ändlig följd. Betrakta till exempel följden s bestående av de 25 första jämna positiva heltalen:

2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,

dvs. $s_n = 2n$ for $n=1, 2, 3, \dots, 25$.

Summan av alla element i denna följd är

$$S=2+4+6+8+10+12+14+16+18+20+22+24+26+28+30+32+34+36+38+40+42+44+46+48+50.$$

Detta långa uttryck är det omöjligt att skriva mer än ett par gånger i en text, och tänk på hur det skulle ha sett ut om vi hade valt att betrakta summan av de 250 första jämna heltalen i stället för de första 25! Vi inför därför ett nytt och kompakt sätt att skriva sådana summor på. Den nya metoden kallas **summanotation** eller **sigmanotation**, och ovanstående summa S skrivs

$$S = \sum_{n=1}^{25} 2n.$$

Denna kompakta notation skall tolkas på följande sätt: För varje heltal n mellan 1 och 25, beräkna och skriv värdena av uttrycket $2n$ och sätt in ett "+" - tecken mellan alla värden.

Generellt skriver vi summan av alla elementen i följderna $(u_n)_{n=1}^t$ som

$$\sum_{n=1}^t u_n.$$

Tecknet $\sum$ är en grekisk bokstav, som motsvarar bokstaven S i det latinska alfabetet, och $\sum$ står för summa. Hela uttrycket utläses som "summan av u_n från $n=1$ till $n=t$ ", och tolkas på följande sätt: För varje heltal n mellan 1 och t , beräkna och skriv värdet av u_n och sätt ett "+" - tecken mellan alla värden.

Betrakta till exempel summan

$$\sum_{n=1}^4 (3n-1).$$

Först beräknar vi värdena $(3n-1)$ för $n=1, 2, 3, 4$:

$$2, 5, 8, 11,$$

och så sätter vi "+"-tecken emellan alla dessa värden:

$$2+5+8+11.$$

Vi har därför

$$\sum_{n=1}^4 (3n-1)=2+5+8+11.$$

Sats B2.1.

Följande två räkneregler gäller för vår nya notation:

$$\sum_{n=1}^m c u_n = c \sum_{n=1}^m u_n,$$

$$\sum_{n=1}^m (u_n + v_n) = \sum_{n=1}^m u_n + \sum_{n=1}^m v_n \text{ där } (u_n)_{n=1}^m \text{ och } (v_n)_{n=1}^m \text{ är ändliga följder av tal och } c \text{ är en konstant.}$$

Bevis.

Dessa räkneregler följer lätt av de kända räknereglerna för 'allmänna' summor:

$$\begin{aligned} \sum_{n=1}^m c u_n &= c u_1 + c u_{i+1} + c u_{i+2} + \dots + c u_m \\ &= c(u_1 + u_{i+1} + u_{i+2} + \dots + u_m) \\ &= c \left(\sum_{n=1}^m u_n \right) \\ &= c \sum_{n=1}^m u_n. \end{aligned}$$

$$\begin{aligned}
\sum_{n=1}^m (u_n + v_n) &= (u_1 + v_1) + (u_{i+1} + v_{i+1}) + \dots + (u_m + v_m) \\
&= (u_1 + u_{i+1} + \dots + u_m) + (v_1 + v_{i+1} + \dots + v_m) \\
&\quad \text{(som ses genom att flytta om element i summan)} \\
&= \sum_{n=1}^m u_n + \sum_{n=1}^m v_n. \quad \square
\end{aligned}$$

Vissa gånger kan man skriva samma summa på mer än ett sätt, detta kallas att **byta variabel** i summan, och detta beskrivs i exempel 2.2.15 (2.2.15) [2.3.17] i [J].

Läs nu om summa- och produktnotation i [J] från och med definition 2.2.12 (2.2.12) [2.3.14] sida 66 (73) [107] till och med exempel 2.2.17 (2.2.17) [2.3.19] sida 68 (78) [108].

4. Strängar

Resten av avsnitt 2.2 (2.2) [2.3] i [J], från och med definition 2.2.18 (2.2.18) [2.3.20] sida 68 (78) [109], handlar om strängar. Vi erinrar om att vi redan har arbetat med strängar i block 1, där vi bevisade, att det finns precis 2^n binära strängar av längd n .

5. Tal

Läs nu avsnitt 2.3 (2.3) [5.2] i [J], det handlar om talsystem.

Grundläggandet för att konvertera ett tal i 10-talssystemet till ett tal i talsystemet med bas N är följande mycket viktiga egenskap vid division med heltal, som går under namnet **Divisionsalgoritmen** :

Sats B2.2 (Divisionsalgoritmen).

Givet ett godtyckligt heltal a och ett positivt heltal N , så existerar det entydigt bestämda heltal k och r (som kallas kvoten och resten) sådana att $a = kN + r$ och $0 \leq r \leq N-1$.

Du känner säkert redan till detta resultat, det säger, att när man dividerar ett heltal med N , så kan man arrangera det så att man alltid får en rest bland talen $0, 1, \dots, N-1$.

Exempel

Tag till exempel $a=23$ och $N=3$, då får man kvoten 7 och resten 2, eftersom $23=7 \cdot 3 + 2$.

Om man dividerar $a=-25$ med $N=3$, så får man kvoten -9 och resten 2, eftersom $-25=(-9) \cdot 3 + 2$. $\square$

Vi skall inte bevisa Divisionsalgoritmen i denna kurs, men du skall känna till resultatet och du skall kunna formulera satsen.

6. Förslag till övningsuppgifter

Övningsuppgifter från [J]

[J] sida 69-71:

Uppg. 4-14, 21-28, 31-36, 43-47, 55-58, 59-62, 83-85, 88-89, 91-92

(sida 79-83: 2, 4, 6, 8, 11, 12, 16-18, 21-22, 24-25)

[sida 110-112: 4-14, 39-46, 51-56, 67-71, 83-86, 87-90, 111-113, 116-117, 119-120]

[J] sida 76:

Uppg. 1-47

(sida 90-91: Uppg. 1-47)

[sida 205: 8-54]

[J] sida 232-233 (265-266) [sida 287]:

Uppg. 1, 2, 4-8, 23-24

Extra övningsuppgifter till Divisionsalgoritmen

Hitta den enligt Divisionsalgoritmen entydigt bestämda kvoten och resten, när a divideras med N , där

1. $a=12$ och $N=7$;
2. $a=-12$ och $N=7$;
3. $a=22$ och $N=3$;
4. $a=-22$ och $N=3$;
5. $a=12$ och $N=3$.

This is the 2nd Edition of the study guide for Block 2 of Algebra and Discrete Mathematics which was written by Pia Heidtmann in 1999. It was translated from Danish and English to the Swedish by Sam Lodin.

The study guide may be printed for *personal use* by anybody with an interest.

This study guide and any parts of it and any previous and future versions of it must not be copied or disseminated in any printed or electronic form or stored on any publicly accessible website other than <http://www.tfm.mh.se/~piahei/adm/res/> without permission from the [author](#).

The author welcomes comments and corrections via [email](#). All contributions incorporated in updates of the manuscript will be acknowledged.

The author would like to thank the following for their contribution to various updates of the original manuscript:
Sarah Norell, Sam Lodin and Fredrik Engström.

Current lecturers:[Pia Heidtmann](#), [Sam Lodin](#).© [Pia Heidtmann](#)

MID SWEDEN UNIVERSITY

Department of Engineering, Physics and Mathematics

Mid Sweden University

S-851 70 SUNDSVALL

Sweden

Updated 070811