Läsanvisning till Discrete matematics av Norman Biggs - 5B1118 Diskret matematik

Mats Boij

28 oktober 2001

1 Heltalen

Det första kapitlet handlar om *heltalen* och deras *aritmetik*, dvs deras egenskaper som har samband med addition och multiplikation. Vi får se *axiomen* för heltalen. För den här kursen är inte dessa axiom så centrala, men det kan vara bra att veta att det finns en solid grund för det vi sedan kommer att hålla på med.

Det handlar också om heltalens ordning och något som kallas *välordning*, som ligger till grund för *induktionsprincipen* och *rekursiva definitioner*. Speciellt det senare har stora anknytningar till datalogi och programkonstruktion.

Vi får reda på begrepp som har med delbarhet att göra, exempelvis *största gemensamma delaren*, och det avslutas med aritmetikens fundamentalsats som säger att faktorseringen av heltal i primfaktorer är unik sånär som på ordningen.

Den viktigaste algoritmen vi stöter på är *Euklides algoritm* och den kommer också att användas flitigt senare i kursen.

Rekommenderade uppgifter

	lättare	svårare
1.3	2	
1.6	2,4(i)	
1.7	1,5	
1.8	1	
1.9	5,6	10,18

1.1 Aritmetik

För att matematiskt beskriva vad som menas med *heltalen* måste man ha några *axiom* som talar om hur heltalen fungerar. Tanken med axiom är att det skall vara en minimal uppsättning regler utifrån vilka alla andra regler och egenskaper skall kunna härledas.

Heltalen beskrivs nu som en mängd $\mathbf{Z}$ med två speciella element 0 och 1 och två räkneoperationer, *addition* och *multiplikation*. Additionen skrivs alltid med $+$, men multiplikationen skrivs ibland med $\times$, ibland med $\cdot$ och ibland helt utan symbol, $ab = a \cdot b = a \times b$.

Översättningar

integer *heltal*
axiom *axiom*
operation *operation*
deduce *härleda*

1.2 Heltalens ordning

Förutom den algebraiska strukturen som heltalen har genom additionen och multiplikationen finns en naturlig *ordning*, som skrivs $\leq$. Det är givetvis den vanliga ordningen som vi lärt oss sedan länge, men för att beskriva denna ordning matematiskt behövs några axiom. Axiom **I8**, **I9** och **I10** är axiomen för en *partialordning*, som är det första exemplet i den här kursen på en *relation*. Senare kommer vi att stöta på en speciell klass av relationer som kallas *ekvivalensrelationer*.

De två senare axiomen, **I11** och **I12** handlar om hur ordningen passar ihop med additionen och multiplikationen.

Det sista axiomet **I13** kallas *välordningsaxiomet* och det är det som gör att vi kan skilja heltalen från *rationella talen*, $\mathbf{Q}$, och från *reella talen*, $\mathbf{R}$.

Poängen med välordningsaxiomet är att en mängd heltal som är nedåt begränsad inte kan innehålla en oändlig strikt avtagande följd. Detta är möjligt för exempelvis de rationella talen där följden $a_n = 1/n$ ligger i mängden av positiva rationella tal som är nedåt begränsad.

Översättningar

ordering relation *ordningsrelation*
arbitrary *godtyckliga*
lower bound *undre gräns*
least member *minsta element*
assert *hävda, påstå*
discrete *diskret*
continuous *kontinuerlig*

Övning 1.2.1 Använd resultatet från uppgift 1.2.1 och axiom **I4** för att visa att $0 \leq 1$.

1.3 Rekursiva definitioner

Här definierar Biggs de *naturliga talen*, $\mathbf{N}$, som de positiva heltalen. Det finns olika konventioner för vad som är naturliga tal. Ofta inkluderas även 0 bland de naturliga talen.

En *rekursiv definition* kan jämföras med en *rekursiv metod* i programmering. För att kunna tala om hur en följd a_n ser ut kan det ibland vara lättare att tala om hur a_n beror på tidigare värden i följden. Detta måste kompletteras med något slags startvärden som säkerställer att vi vet hur följden börjar. Antalet startvärden beror på hur rekursionsformeln ser ut. Om a_n hela tiden beror på de tre tidigare värdena behövs till exempel tre startvärden, a_0, a_1 och a_2.

1.4 Induktionsprincipen

Induktionsprincipen som också togs upp i kursen 5B1115 Matematik I, har mycket gemensamt med rekursiva definitioner. Exempelvis kan man säga att det är induktionsprincipen som gör att vi säkert vet att en rekursiv definition verkligen definierar en följd.

I det här avsnittet formuleras induktionsprincipen väldigt abstrakt som Sats 1.4. Här nämns inget påstående eller liknande utan bara en mängd av heltal. Det är viktigt att förstå att detta inte är något annat än den vanliga induktionsprincipen.

I slutet av avsnittet nämns begreppet *stark induktion*, som är en mer generell form av induktion än den som vi har sett tidigare. Det är en form av induktion som vi kommer att få användning för senare, exempelvis när det gäller primtalsfaktorisering. Skillnaden är att vi inte går från n till $n + 1$ som vi i tidigare gjort, utan att vi kan anta att vi vet att det aktuella påståendet är sant för *alla* naturliga tal mindre än n och att vi använder den kunskapen för att visa att det då också gäller för n. Styrkan ligger alltså i att vi inte bara använder påståendet för ett av de föregående värdena, utan för alla.

Översättningar

induction basis *induktionsbas*
induction hypothesis *induktionsantagande*

1.5 Kvot och rest

Sats 1.5 kallas ofta *divisionsalgoritmen*, även om det egentligen inte är fråga om en algoritm, utan bara är en sats om *existens* och *entydighet* av kvot och rest vid division av ett heltal med ett positivt naturligt tal.

Representation i olika baser.

De algoritmer för division, addition och multiplikation vi har lärt oss i skolan bygger alla på det så kallade *positionssystemet*. Det vi inte så ofta har tänkt på är kanske att dessa algoritmer som vi har lärt oss för det *decimala* positionssystemet också fungerar om vi inte använder just *basen* 10. I det *binära* systemet använder man istället basen 2, i det *oktala* basen 8 och i det *hexadecimala* basen 16. I allmänhet kan vi använda ett godtyckligt heltal större än ett som bas för ett positionssystem och algoritmerna fungerar på samma sätt.

För att bestämma *siffrorna* i ett tal i en viss bas används divisionsalgoritmen upprepade gånger. Genom att ta resten vid division med b av talet får man den sista siffran. Gör vi samma sak med den kvot vi så fick får vi den näst sista siffran och så vidare.

Det kan vara värt att notera att siffrorna som används i basen 16 är

$$0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E.$$

Översättningar

quotient *kvot*
remainder *rest*

1.6 Delbarhet

I det här avsnittet definieras ett antal viktiga begrepp när det handlar om *delbarhet*. I dagligt tal kan man ibland säga "jämt delbar med" istället för "delbar med", men det betyder precis samma sak.

Översättningar

divisor *delare*
factor *faktor*
divides *delar*
divisible *delbar*
multiple *multipel*

1.7 Största gemensamma delare

Den största gemensamma delaren, d, mellan två heltal, a och b, defineras här genom tre egenskaper

1. $d|a$ och $d|b$, dvs d delar både a och b.
2. Om $c|a$ och $c|b$ så $c|d$, dvs alla heltal som delar både a och b delar också d.
3. $d \geq 0$.

Från namnet att döma vore det kanske naturligare att definiera det som det största naturliga tal som delar både a och b, men den definition som ges här är lite mer allmän, och den kan framför allt användas i andra fall än för heltalen, tex för polynom.

En viktig konsekvens av Euklides algoritm är Sats 1.7 som säger att den största gemensamma delaren mellan a och b kan skrivas som $ma + nb$ för några heltal m och n. Detta kommer att vara till stor användning senare. Euklides algoritm använd baklänges gör det möjligt att hitta sådana m och n.

Översättningar

greatest common divisor *största gemensamma delare*
gcd *sgd*
Euclidian algorithm *Euklides algoritm*
coprime *relativt prima*

1.8 Primtalsfaktorisering

Vi får en defintion av *primtal*. Det är viktigt att notera att 1 inte är ett primtal, även om det uppfyller kravet att de enda delarna är 1 och p.

Argumentet för att det finns en primtalsfaktorisering för varje heltal bygger på välordningsaxiomet och är en variant av stark induktion. Det är en väldigt vanlig typ av argument:

1. Om det inte gäller för alla positiva heltal måste det finnas ett minsta positivt heltal för vilket det inte gäller. Låt m vara detta heltal.

2. Använd att det nu gäller för alla mindre heltal för att visa att det faktiskt ändå gäller för m.

3. Vi har kommit till en motsägelse och alltså måste antagandet om att det finns heltal för vilka det inte gäller vara falskt, och vi drar slutsatsen att det är sant för alla positiva heltal.

Sats 1.8.2 kallas *aritmetikens fundamentalsats* och säger att primtalsfaktorisering är *unik*, eller *entydig*, sånär som ordningen av faktorerna.

Beviset för aritmetikens fundamentalsats bygger på *hjälpsatsen*, Sats 1.8.2, som säger att ett primtal som delar en produkt av tal måste dela minst en av faktorerna.

Översättningar

prime *primtal*
prime factorization *primalsfaktorisering*