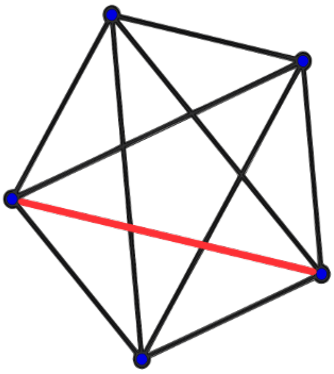
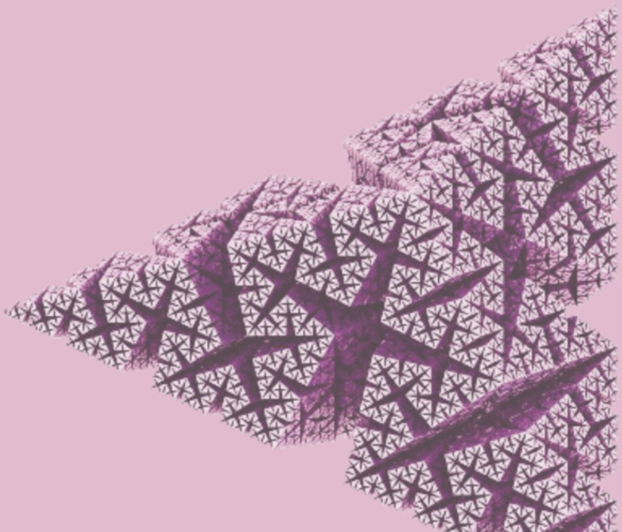
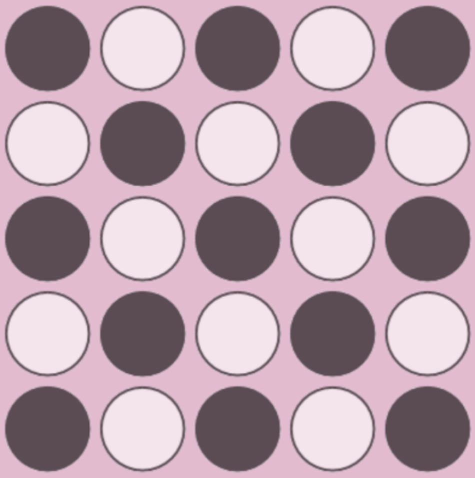
Mattekollo 2019



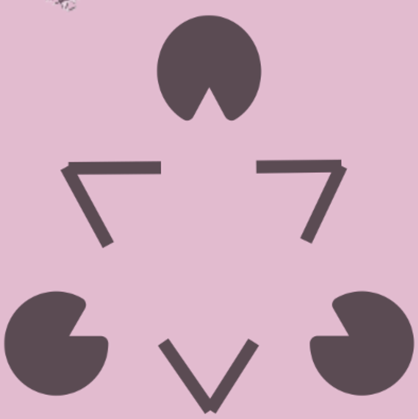
Lektionsmaterial

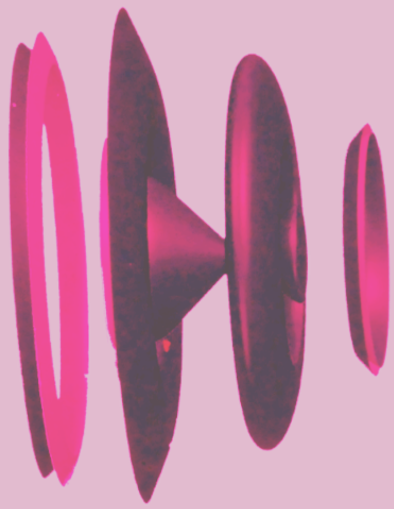
åk 9-gy2





$$\frac{31\varepsilon}{16}$$





Lektionsmaterial Mattekollo 2019

åk 9-gy2

E. Arctaedius, J. Björkund, V. Chapovalova, S. Fodor, M. Hellström och U. Lundström

Norrtälje, augusti 2019

Layout: V.Chapovalova

Typografi: E. Lundell

Innehåll

Lektionsmaterial blå gruppen

1. Vända brickor I

2 augusti

1. Lägg sju brickor på rad som på bilden nedan.

●○○○●●○

I ett drag får du vända på två intilliggande brickor. Kan du utföra drag och till slut få alla brickor med svart sida upp?

2. Givet positionen ●○○○○●●●○, är det möjligt att till slut få alla brickor med vit sida upp? svart sida upp?

i) Giltiga drag är att vända på två intilliggande brickor.

ii) Giltiga drag är att vända på två vilka som helst brickor.

iii) Giltiga drag är att vända på tre brickor som ligger på rad intill varandra.

3. Lägg fyra brickor på rad, alla med svart sida upp. I ett drag får du vända på vilka tre brickor du vill. Kan du utföra drag och till slut uppnå att alla brickor ligger med vit sida upp?

4. Lägg istället fem brickor på rad, alla med svart sida upp. I ett drag får du vända vilka fyra du vill. Kan du göra så att alla brickor ligger med vit sida upp?

5. Tänk att n brickor ligger med svart sida upp, där n är ett positivt heltal. I ett drag vänds $n-1$ brickor, vilka som helst. Går det att till slut få alla med vit sida upp?

6. Lägg 25 brickor schackmönstrigt i en 5×5-kvadrat. I ett drag väljer du tre

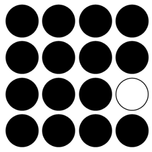
●○●○●
○●○●○
●○●○●
○●○●○
●○●○●

brickor som ligger på rad, antingen i en rad eller kolumn, som du sedan vänder på. Kan du göra 5×5-kvadraten helt svartfärgad? Kan du göra den helt vitfärgad?

7. Lägg 16 brickor schackmönstrigt i en 4×4-kvadrat. I ett drag väljer du tre brickor som ligger på rad, antingen i en rad eller kolumn, som du sedan vänder på. Kan du göra 4×4-kvadraten enfärgad?

8. Återigen, lägg 16 brickor schackmönstrigt i en 4×4-kvadrat. I ett drag får du vända på alla brickorna i godtycklig $2 \times 2-$kvadrat och 3×3-kvadrat. Kan du utföra drag och till slut få en enfärgad kvadrat?

9. Lägg 16 brickor så som i figuren. Kan du göra brädet helt svart? I ett drag får du



- vända på en hel rad
- vända på en hel kolumn
- vända på huvuddiagonalen eller antidiagonalen
- vända på något parallellt med huvuddiagonalen eller antidiagonalen.

2. Vinkeljakt

2 augusti

Matematiker höll på med geometri redan under antika Grekland, de viktigaste verken inom geometri är Euklides böcker "Elementa" som skrevs runt år 300 f.Kr. och innehåller viktiga definitioner och bevis. Som all matematik baseras geometri på axiom, påståenden som inte bevisas utan antas vara sanna. Dock måste alla andra påståenden efteråt kunna härledas från dessa axiom. Inom geometri använder vi dem axiom som Euklides använde i Elementa, dock är det knappast förväntat att härleda alla geometriska bevis från dessa. När ni löser följande uppgifter kan ni till exempel fritt använda att vinkelsumman i en triangel är 180° eller att en likbent triangel har två lika vinklar.

När man löser en uppgift i geometri lägger man oftast fokus på vinklar. Vinklar är bekväma att jobba med tack vare satser om parallella linjer och vinkelsummor (till exempel i en triangel). Metoden heter vinkeljakt och iden är att översätta all given information till vinklar och sedan komma fram till en lösning genom att räkna på olika vinklar.

1. I en triangel är $2 \cdot \angle A = 3 \cdot \angle B = 6 \cdot \angle C$. Vad är triangelns vinklar?

2. I fyrhörningen $ABCD$ är diagonalerna rätvinkliga. Vidare är vinklarna $\angle ACD = 30^\circ$, $\angle BDA = 40^\circ$ and $\angle DBC = 50^\circ$. Räkna ut vinkeln $\angle BCD$.

3. Låt $AB = AC$ i triangeln ABC. Bisektrisen från B delar triangeln i två likbenta trianglar. Vad är triangelns vinklar?

4. Låt $AB = AC$ i triangeln ABC. Låt BE vara en av höjderna. Visa att $2 \cdot \angle CBE = \angle BAC$.

5. Visa att hypotenusan är dubbelt så lång som en av kateterna i en rätvinklig triangel om och endast om en av vinklarna är $30°$.

6. $ABCD$ är en symmetrisk parallelltrapets där AB och CD är parallella och $AD = BC$. Visa att om AC är bisektrisen vid $\angle A$ så är $AB = BC$.

7. Låt AB vara en sträcka med mittpunkt M och låt C vara en punkt i planet. Visa att om $\angle CMA = 90°$ så är $CA = CB$.

8. Låt AB vara en korda i en cirkel med mittpunkt O. Visa att linjen l som går igenom O och är rätvinklig mot AB delar kordan i två lika stora delar.

9. Låt linjen l tangera en cirkel med mittpunkt i O i punkten A. Visa att OA och l skär varandra i en rät vinkel. (Tips: Visa att om linjerna inte skär varandra i en rät vinkel kan l inte tangera cirkeln.)

10. Linjen l tangerar en cirkel med mittpunkt O vid punkten A. Kordan BC är parallell med l. Visa att $AB = AC$.

11. Låt A, B och C ligga på en cirkel med mittpunkt O. Visa att $\angle ABC = 90°$ om och endast om AC är en diameter av cirkeln.

12. I triangeln ABC är $\angle A = 90°$ och $\angle B = 15°$. Låt AD vara en av höjderna. Visa att $BC = 4 \cdot AD$.

3. Spelteori I

3 augusti

Du och en kompis spelar ett enkelt spel. Ni har två högar med stenar; tre stenar i den ena högen och en enda i den andra. Ni turas om att göra ett varsitt drag. Ett drag går ut på att välja en hög och ta minst en sten från den högen (man kan ta fler om man vill). Man måste alltid ta minst en sten, och får aldrig ta från flera högar i samma drag. Vinner gör den som tar den sista stenen.

1. Vem kommer att vinna, du eller din kompis? Hur då; vilka drag gör ni? (Anta att både du och din kompis spelar så bra som ni bara kan.)

2. Efter att ha löst den förra uppgiften känns spelet orättvist. Ni väljer att istället börja med två högar med fem stenar i varje. Vem vinner nu?

3. Vad händer om ni har två högar, båda med lika många stenar i?

4. Utifrån detta, vem vinner om ni har en hög med 6 stenar och en med 7?

5. Vem vinner om vi har två högar med valfritt antal stenar n och m? (Ledtråd: anta att $m > n > 0$; fallet $n = m$ har ni redan löst)

6. Bra jobbat! Då har vi löst alla fall med två högar. Men vad händer om vi lägger till fler högar? Undersök fallet när vi har tre högar: två med n stenar och en med m. Vem vinner nu?

Notation. Vi inför följande notation: ett $(a_1, a_2, a_3, \cdots, a_n)$-spel är ett spel med n högar, där hög nummer i har a_i stenar. T.ex. är det första spelet vi undersökte (med 1 respektive 3 stenar i högarna) ett $(1,3)$-spel (eller $(3,1)$-spel; ordningen spelar ingen roll).

7. **(a)** Vem vinner ett $(2,2,7,7)$-spel?

   (b) Vem vinner ett $(3,3,4,5)$-spel?

   (c) Vem vinner ett $(3,3,3,3,4,5)$-spel?

   (d) Vem vinner ett $(3,3,4,5,4,5)$-spel?

Undersök nu den klass av spel som har högar med tvåpotenser:

8. Vem vinner ett $(1,2,4)$-spel?

9. Vem vinner ett $(1,2,4,8)$-spel?

10. Hitta en strategi som kan vinna vilket spel $t_n = (1,2,4,\cdots,2^n)$ som helst.

Extrauppgifter

11. Anpassa er strategi till spel $t'_n = (2,4,\cdots,2^n)$ (dvs. t_n med ett-högen borttagen).

12. Hitta en strategi för spelen $t''_n = (1,1,2,4,\cdots,2^n)$. Jämför hur strategin påverkas av att lägga till en ett-hög och att ta bort ett-högen.

13. Spelar det någon roll om det är ett-högen vi tar bort eller dubblerar? Vad händer om vi istället tar 2-högen etc.?

14. Vad händer om vi lägger till fler kopior av en hög vi redan har? T.ex. $(1,2,2,2,4)$?

15. Hitta den vinnande strategin för ett godtyckligt tvåpotens-spel (dvs. spel där alla högar är en tvåpotens).

16. Hur kan ni passa in högar som inte är tvåpotenser? Försök att anpassa er strategi så att ni kan lösa t.ex. $(1,2,4,5)$-spelet. (Ledtråd: hur ser femman ut skrivet som tvåpotenser?)

4. Randvilkel och Tangetkorda

3 augusti

1. A, B och C ligger på en cirkel med mittpunkt O så att O och C ligger på samma sida av linjen AB. Visa att $\angle AOB = 2 \cdot \angle ACB$.

2. A, B och C ligger på en cirkel med mittpunkt O så att O och C ligger på olika sidor av linjen AB. Visa att $\angle AOB = 2 \cdot (180 - \angle ACB)$.

3. Låt A, B, C och D ligga på en cirkel så att C och D ligger på samma sida av linjen AB. Visa att $\angle ACB = \angle ADB$.

4. Låt A, B, C och D ligga på en cirkel så att B och D ligger på olika sidor av linjen AC. Visa att $\angle ABC + \angle ADC = 180°$.

Satserna ovan kallas för randvinkelsatsen och är oerhört viktiga. Den låter oss flytta runt vinklar längst en cirkel, om vi kan räkna ut randvinkeln av en korda AB vet vi sedan $\angle APB$ för alla punkter P på cirkeln. Fallet där AB är en diameter av cirkeln och $\angle APB = 90°$ är ett särskilt viktigt fall av denna sats, de första uppgifterna behandlar mest detta fall.

5. Låt AB vara en korda av en cirkel med mittpunkt O. Visa att cirkeln med diameter AO delar AB i två lika delar. (Tips: tänk på uppgift 8 från vinkeljakts uppgifterna.)

6. Låt P vara en punkt innanför cirkeln ω. Visa att mittpunkterna av de kordorna som innehåller P ligger på en cirkel.

7. Låt AB vara en diameter av en cirkel och låt AC vara en korda. Förläng kordan till punkten D så att $CD = AC$. Visa att triangeln ABD är likbent.

8. Låt cirklarna ω_1 och ω_2 skära varandra i punkterna A och B. Låt AC och AD vara diametrar av cirklarna. Visa att B, C och D ligger på en linje.

9. Låt $ABCD$ vara en parallelltrapets inskriven i en cirkel där AB och CD är parallella. Visa att $AD = BC$.

10. Låt $ABCDE$ vara en femhörning så att $BCDE$ är en kvadrat med mittpunkt O och så att $\angle A = 90°$. Visa att AO är en bisektris av $\angle BAE$.

11. Fyrhörningen $ABCD$ ligger på en cirkel med centrum O så att $AOB = 2\alpha$ och $COD = 2\beta$. Låt linjerna AC och BD skära varandra i punkten E. Räkna ut vinkeln $\angle AEB$.

12. Låt triangeln ABC ligga på en cirkel ω. Höjderna från B och C skär cirkeln i D och E. Visa att $AD = AE$.

13. Cirklarna ω_1 och ω_2 skär varandra vid A och B. Låt P vara en godtycklig punkt på ω_1 och låt linjerna PA och PB skära ω_2 vid punkterna C och D. Visa att längden av CD är oberoende av valet av punkten P.

14. Låt AB vara en korda av en cirkel med mittpunkt O så att $\angle AOB = 2\alpha$. Låt linjen l tangera cirkeln vid punkten A. Visa att AB och l skär varandra med en vinkel α.

Satsen ovan kallas tangentkorda-satsen och hjälper oss med att hantera tangenter vid vinkeljakt. En intressant tanke är att se tangentkorda satsen som ett specialfall av randvinkelsatsen. Om vi låter punkten C i randvinkelsatsen gå mot punkten A längst cirkelns rand kommer linjen AC att gå mot tangenten vid A.

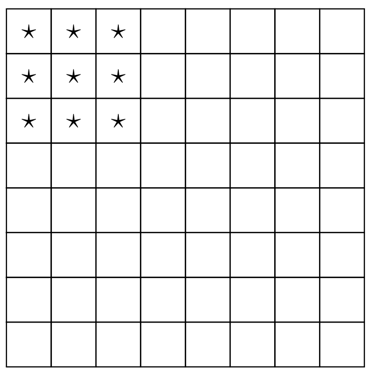
15. Låt triangeln ABC vara inskriven i en cirkel. Tangenten vid A skär linjen BC vid punkten D. Räkna ut $\angle ADB$ givet att $\angle B = \beta$ och $\angle C = \gamma$.

16. Cirklarna ω_1 och ω_2 skär varandra i punkterna A och B. Låt l vara godtycklig linje genom A. Denna linje skär cirklarna i punkterna C och D, tangenterna vid C och D skär varandra vid E. Visa att $\angle CED$ är oberoende av valet av linjen l.

5. Vända brickor II

4 augusti

1. I övre vänstra hörnet på ett 8×8-bräde sitter nio gräshoppor lugnt och fridfullt i nio små rutor, se figur 5.1. Dessa gräshoppor önskar flytta till det förlovade



Figur 5.1: Nio små gräshoppor.

landet i öster. I varje flytt deltar precis två gräshoppor, den ena står still och blir hoppad över, den andra hoppar och landar symmetriskt kring den som den hoppade över. Alla symmetriska drag är godkända så länge de hamnar inom brädet samt så länge det som mest står en gräshoppa i en ruta.

Kan gräshopporna flytta österut till den övre 3×3-delkvadraten i högra hörnet? Kan de flytta sydöst den undre?

2. Lägg 25 brickor i en 5×5-kvadrat alla med svart sida upp. Vänd på en av brickorna, vilken du vill. Denna bricka kallar vi för *startbrickan*. Från denna bricka vänd en till bricka på *springaravstånd*. Upprepa. Kan du till slut ha gjort kvadraten enfärgad *och* även nå startbrickan från den bricka du sist vände?

3. Lägg brickor schackmönstrigt i en 4×4-kvadrat. I ett drag får två intilliggande brickor vändas. Kan kvadraten bli enfärgad?

4. Lägg brickor schackmönstrigt i en 4×4-kvadrat. I ett drag får tre intilliggande brickor på samma rad eller kolumn vändas. Kan kvadraten bli enfärgad?

5. Tänk er att ni har brickor schackmönstrigt utplacerade i en 8×8-kvadrat. I ett drag får tre intilliggande brickor på samma rad eller kolumn vändas. Kan kvadraten bli enfärgad?

6. Uppskattningar och uppdelningar

4 augusti

1. Kan man sätta ut stjärnor i vissa rutor på en 10×12-tabell (10 rader och 12 kolumner) på så sätt att det finns exakt 7 stjärnor i varje rad och exakt 5 stjärnor i varje kolumn?

2. Fyra vänner kom hem från att ha fiskat. Varje par av vänner räknade ihop hur många fiskar de hade fått tillsammans. Man fick följande tal: 7, 9, 14, 14, 19, 21. Kan man lista ut hur många fiskar var och en av vännerna fiskade upp?

3. I en $3x3$-kvadrat står det tal i rutorna. Summan av alla talen är lika med 100.

(a) Visa att man kan ställa ut tre torn i kvadraten som inte hotar varandra så att summan av talen på rutorna de står på är åtminstone 33.

(b) Visa att om alla talen är heltal, så kan man ställa de tre tornen på så sätt att summan av talen de står på är åtminstone 34.

4. På en cirkel står 100 tal. Summan av alla talen är lika med 1. Kan summan av tre på varandra följande tal i cirkeln alltid vara negativ?

5. På en cirkel står alla heltalen från 1 till 23, kanske i oordning. Visa att det finns tre tal som följer efter varandra på cirkeln vars summa är åtminstone 33.

6. 7 barn i ett volleybollag väger 332 kg tillsammans (man behöver inte väga ett helt antal kilon). Visa att man kan välja tre barn som tillsammans väger minst 142 kg.

7. På Erkenlaboratoriet finns det 20 medarbetare. Varje morgon finns det några par av dem som skakar hand med varandra, kanske olika par olika morgnar. På en månad skedde exakt 2019 handskakningar. Visa att man kan välja en grupp med 7 medarbetare så att det inom gruppen har skett åtminstone 224 handskakningar.

8. Det finns 10 soldater i en trupp. Var och en av de 100 dagarna de var i tjänst så var det fyra av dem som ansvarade för att laga middag. Visa att det finns ett par soldater som lagade middag ihop åtminstone 14 gånger.

9. Det finns två halsband med 100 vita och 100 svarta pärlor i varje. Stina vill lägga ena halsbandet ovanpå det andra (det går bra att vända och vrida halsbanden) på så sätt att så många av pärlorna som möjligt av samma färg sammanfaller. Vilket är det största antalet sammanfallande pärlor som Stina kan garantera?

Extrauppgifter

10. På en cirkel står talen 1, 2, 3, ..., 10 i någon ordning. Eskil räknade ut 10 summor utav alla tripplar av tal som kommer efter varandra och skrev upp den minsta summan på tavlan. Vilket är det största möjliga talet som kan stå på tavlan?

11. Rutorna på en $n \times n$ - tabell är fyllda med talen 1 till n och det finns n kopior av varje tal. Visa att det finns en rad eller kolumn som innehåller åtminstone $\sqrt{n}$ olika tal.

7. Spelteori II

5 augusti

Fångarnas Dilemma. Du och din vän har rånat en bank tillsammans. Polisen vill att du ska tjalla på din kompis. Om du gör det så lovar de att hen får hela skulden, och kommer dömas till fängelse i åtta år, medan du kommer gå fri. Du antar att polisen ger din kumpan samma erbjudande, och att om ingen av er skvallrar så kommer ni båda dömmas till ett år i fängelse. Om ni båda skvallrar så kommer ni båda få 6 år var i finkan.

1. Du och din vän kan inte kommunicera med varandra. Du vet inte vad hen kommer göra.

(a) Vad ska du göra för att se till att du hamnar i finkan så kort tid som möjligt (men utan att bry dig om hur länge din kompis sitter där)? Spelar det någon roll vad din kumpan väljer att göra?

(b) Anta att din vän också kommer försöka minimera sin egen fängelsetid. Hur kommer spelet då sluta?

2. Du blir arg och kommer på ett tredje alternativ - du skulle kunna spotta polisen i ansiktet. I så fall kommer du få 11 år i fängelse oavsett vad din kumpan väljer, och hen får samma straff som om du inte skvallrade. Ändrar det här resultatet av spelet? Varför eller varför inte?

Resultat-matris. Ett bra hjälpmedel när man analyserar spel som dess är en resultatmatris. Detta är en tabell där kolumnerna motsvarar ena spelarens alternativ och raderna motsvarar den andra spelarens alternativ. I varje ruta skriver man in hur många poäng/straff varje spelare får om de tar dessa alternativ. T.ex. är resultatmatrisen för Fångarnas Dilemma:

	skvallra inte	skvallra
skvallra inte	1, 1	8, 0
skvallra	0, 8	6, 6

Definition 1. En *strategi* är en beskrivning av alla alternativ en spelare kommer att välja i ett spel. (Detta är så kallade *rena* strategier - blandade strategier definieras nedan.)

3. Modifiera matrisen till Fångarnas Dilemma genom att lägga till nya rader och/eller kolumner så att utfallet blir ett annat!

Gissa krona. I spelet *Gissa Krona* så har den första spelaren två strategier - antingen väljer hen *krona* eller *klave*. Den andra spelaren ska nu gissa vilken strategi den första spelaren valde. Om hen gissar rätt vinner hen en krona från den första spelaren - om andra spelaren gissar fel får hen istället ge en krona till första spelaren.

Definition 2. En strategi är *dominant* om den aldrig ger spelaren ett sämre resultat än alla andra strategier hen kan välja, oavsett vad den andra spelaren väljer. En *strikt* dominant strategi ger alltid strikt bättre resultat än de andra möjliga strategierna.

4. Visa att det inte finns några dominanta strategier i sten-sax-påse!

5. Finns det några dominanta strategier i Gissa Krona eller Fångarnas Dilemma?

Dejtkampen. Du ska på dejt i kväll, men ni har inte bestämt vart ni ska gå och du har glömt din mobil. Du går helst på bio och din dejt går helst till en nöjespark. Om ni båda går till samma ställe så kommer den som helst ville gå dit få 4 roliga timmar, den andra får 1. Om ni går till olika så får ni inga kul timmar alls.

6. Har Dejtkampen några dominanta strategier?

> **Definition 3.** En *Nashjämvikt* uppstår i ett spel när alla deltagare har valt en strategi, och ingen deltagare kan vinna något på att byta strategi själv (dvs. utan att någon annan byter strategi).

7. Har följande spel någon eller några Nashjämvikter? Vilka i så fall?

(a) Fångarnas Dilemma?

(b) Gissa Krona?

(c) Dejtkampen?

(d) Sten-sax-påse?

Blandade strategier. I en *blandad strategi* har en spelare givet en viss chans till varje ren strategi. T.ex. kanske fången väljer att skvallra med sannolikhet $\frac{2}{5}$ och att inte skvallra med sannolikhet $\frac{3}{5}$.

8. Vi ska nu analysera Gissa Krona och försöka hitta en Nashjämvikt:

(a) Antag att gissaren i gissa krona har en blandad strategi där hen väljer krona med sannolikhet p. Vilken strategi ska du då välja för att maximera din genomsnittliga vinst?

(b) Givet det ovan, vilket värde på p kommer din motståndare välja?

(c) Vilket värde ska du välja på din sannolikhet q för att välja krona för att inte motspelaren ska kunna vinna mer än 50% av gångerna?

(d) Finns det någon Nashjämvikt här med blandade strategier?

9. Bevisa:

(a) om båda spelare har en strikt dominant strategi så har spelet precis en Nashjämvikt.

(b) ett spel kan ha en Nashjämvikt utan att någon spelare har en dominant strategi. (Skapa ett exempel!)

10. Antag att vi från ett spel tar bort en dominerad strategi i taget. Bevisa att om det bara finns en strategi per spelare kvar i slutet så motsvarar de en Nashjämvikt i det ursprungliga spelet (dvs. spelet innan vi började ta bort strategier).

11. Vi modifierar Dejtkampen genom att ge dig ett nytt alternativ: innan ni väljer vart ni går kan du välja att minska ditt antal kul timmar med 2, oavsett var ni går. Din dejt får veta om du har valt att sabba innan hen väljer vart den ska gå.

(a) Hur många rena strategier får varje spelare nu?

(b) Hitta en Nashjämvikt i det här spelet när ni båda använder rena strategier.

12. Använd förhållandet mellan hur kul du har de i de olika fallen för att hitta en Nashjämvikt med blandade strategier i Dejtkampen.

8. Matematisk yatzy

5 augusti

Kombinatorik

1. (10) Borgmästaren bestämde att 2019 hus i ett område skulle målas om så att åtminstone 1000 hus är röda och åtminstone 1000 är gröna. Hur många olika färger kan en målare som mest använda för att slutföra uppdraget? (Ett hus har endast en färg).

2. (20) På hur många sätt kan en 3x5-rektangel delas upp i 1x3-rektanglar? (Sätten som man får från varandra genom att behöva rotera eller spegla rektangeln räknas som olika.)

3. (30) En 4x4-tabell fylls upp med hjälp av Minröj-regler, dvs minor placeras i vissa ruta och sedan får varje tom ruta en siffra som talar om antalet minor som finns intill rutan (endast ortogonalt). Vilken är den största möjliga summan på dessa siffror som man kan få?

4. (40) I ett bostadshus finns det n bostadsrätter och totalt bor det n katter och n hundar i huset. Varje bostadsrätt har olika uppsättningar djur (om man bara ser till antal och sort). Vilket är det största möjliga värdet på n?

5. (50) Simon och Selma skriver upp fyrsiffriga tal. Simon skriver upp tal där första siffran är lika med summan av de övriga tre, Selma skriver upp tal där sista siffran är lika med summan av de övriga tre. Vem kommer att skriva upp fler tal och hur många fler?

Textuppgifter

6. (10) Kackerlackan Egon deklarerar att han kan springa med hastigheten 50 m/min. Det visade att ingen trodde på honom och det var med rätta, för att han hade blandade ihop det. Han trodde att meter var 60 cm och att en minut bestod av 100 sekunder. Vilken hastighet (i vanliga m/min) har kackerlackan Egon?

7. (20) En snickare sågade upp ett schackbräde i rutor på 70 minuter. Hur lång tid tar det för honom att såga upp ett likadant bräde i 2x2-kvadrater?

8. (30) En 3x3-tabell fylldes med talen 1 till 9 (varje tal förekom en gång) och man räknade ut alla rad- och kolumnsummor. Vilket är det största möjliga antalet på varandra följande tal som kan finnas bland dessa summor?

9. (40) Erland och Micke har n mynt var. Varje mynt är värt 1 cent, 50 cent eller 1 euro. Det visar sig att båda har lika mycket pengar men inte en likadan uppsättning mynt. Vilket är det minsta n så att detta är möjligt?

10. (50) I en regional tävling deltar 16 volleybollag. Var par av lag möter varandra två gånger. Det är de 8 bästa lagen som går vidare till SM. Man ordnar lagen efter antalet vinster och om vissa har vunnit lika många så är det slumpen som avgör i vilken ordning de kommer. Det finns inget oavgjort i volleyboll. Vilket är det minsta antalet vinter som garanterar att man tar sig vidare till SM?

Algebra

11. (10) För talen a och b gäller $(a+b)(a+b-1) = ab$ och $a^2 - b^2 = 3$. Bestäm värdet av $a^3 - b^3$.

12. (20) För heltalen a, b, c och d gäller att $\frac{a-b}{c-d} = \frac{a+b}{c+d}$ och $ad = 60$. Bestäm produkten av alla de fyra talen.

13. (30) Det finns tre olika tal a, b, c sådana att likheterna $a^2 - bc = b^2 - ac = c^2 - ab$. Bestäm alla värden som uttrycket $\dfrac{a}{b+c} + \dfrac{b}{a+c} + \dfrac{c}{a+b}$ kan anta.

14. (40) Bestäm alla lösningar till ekvationssystemet $\begin{cases} |x| + |y| = 5 \\ ||x| - y| = 1 \end{cases}$

där $|x|$ står för absolutbeloppet av talet x.

15. (50) För de positiva talen x, y, z gäller det att $xyz = 1$, $x + \frac{1}{z} = 5$, $y + \frac{1}{x} = 29$. Bestäm $z + \frac{1}{y}$.

Heltal

16. (10) Bestäm det minsta positiva heltalet n för vilket talet $(n+1)(n+2)(n+3)(n+4)$ är delbart med 1000.

17. (20) Hitta alla rektanglar med heltalssidor vars area har samma värde som dess omkrets.

18. (30) Några positiva heltal multiplicerades med varandra och man fick svaret 224. Det minsta av talen var hälften så stort som det största. Hur många tal multiplicerades?

19. (40) Bestäm tre olika positiva heltal där ingen är större än 16, så att summan av deras inversa värden är lika med $\frac{1}{2}$.

20. (50) Hur många positiva heltal n mindre än 1000 finns det sådana att $2^n - n^2$ är delbart med 7?

Geometri

21. (10) Dela upp en 1x1-kvadrat i sju rektanglar som alla har omkretsen 2.

22. (20) I en likbent triangel är bisektrisen till en av basvinklarna lika lång som basen. Bestäm triangelns vinklar (i grader).

23. (30) På ett rutat papper har man markerat fyra gitterpunkter som markerar hörnen på en 4x4-kvadrat. Markera två gitterpunkter till och förbind dem med en självslutande bruten linje så att sexhörningen som bildas har den minsta möjliga arean av alla.

24. (40) I triangeln ABC har man valt punkten K på sidan BC så att $\angle BAK = 24°$. På sträckan AK har man markerat punkten M så att $\angle ABM = 90°, AM = 2BK$. Bestäm vinkeln B.

25. (50) Ett snöre viktes ihop i tredjedelar, sedan viktes den ihop till tredjedelar igen. Därefter klippte man upp snöret utan att klippa på någon punkt där man vikte. Snören föll isär i bitar, varav två har längderna 2 cm respektive 6 cm. Bestäm hur långt snöret kunde ha varit från början?

9. Spelteori III

7 augusti

Under den här lektionen antar vi att båda spelarna alltid spelar så bra de kan. Alltså när jag frågar "vem vinner" så menar jag *vem kan vinna, oavsett vad motståndaren gör?*

Vattenfall. Spelet *Vattenfall* spelas på ett rutnät med rader och kolumner. Varje ruta kan vara antingen tom eller ha en markör i sig. Två spelare turas om att göra ett drag - den första spelaren som inte kan göra ett drag förlorar.

När en spelare ska göra ett drag börjar hen med att välja en markör. (Finns det ingen markör kvar har hen förlorat.) Hen tar bort den valda markören och går sedan igenom alla rutor under den borttagna markören i samma kolumn. I varje ruta får hen välja om den vill låta rutan vara oförändrad, eller ta bort den markör som ev. ligger där eller, om rutan är tom, lägga dit en ny markör.

1. Vem vinner följande speluppsättningar (x är markörer):

(a)

x	
	x

(b)

x	x

(c)

	x	
x	x	
	x	x

2. Antag att när spelet börjar ligger bara två markörer på planen.

(a) Vem vinner om ni lägger dem i samma kolumn?

(b) Skapa en speluppsättning med två kolumner som första spelaren vinner.

(c) Skapa en speluppsättning med två kolumner som andra spelaren vinner.

(d) Formulera en regel som berättar vem som kommer vinna ett spel som börjar med bara två markörer.

3. Antag att vi har ett spel där alla n markörer ligger i samma rad.

(a) Vem vinner?

(b) Antag att vi lägger till en markör i en annan rad i någon av de n kolumnerna. Hur påverkar det vem som vinner?

(c) Vad händer om vi istället lägger den nya markören i en ny kolumn?

4. Säg att vi i början har n markörer i en rad och m markörer i en annan samt att ingen kolumn har två markörer. Vem vinner då?

5. Antag istället att vi har ett spel med n markörer, alla i samma rad. I $m < n$ kolumner som redan har en markör lägger vi en till markör på en nya rad, så att på en alla de nya markörerna hamnar i samma rad. Vem vinner då?

6. Om det enda ni vet om ett spel är att det i varje rad finns ett jämnt antal markörer, kan ni då säga vem som vinner?

7. Om det är så att alla rader utom en har ett jämnt antal markörer, vem vinner då?

8. Bevisa att om det finns några rader med ett udda antal markörer så kan man alltid i ett enda drag göra så att alla rader får ett jämnt antal markörer!

9. Vilken är den vinnande strategin för Vattenfall?

Extrauppgifter

10. Vi inför följande notation för Vattenfall:

1. En kolumn beskrivs uppifrån och ner med 0 om rutan är tom och 1 om den har en markör.

2. En tabell beskrivs genom att beskriva kolumnerna och sätta komman mellan dem.

T.ex. blir tabellen från 1a. “100,001”.

(a) Hur beskrivs tabellen från 1b?

(b) Hur beskrivs tabellen från 1c?

11. Hur beskrivs ett spel med n markörer, alla i samma rad?

12. Hur skulle man kunna översätta en “Vattenfalls-beskrivning” till de spelbeskrivningar vi använde 3 augusti? Hur skulle t.ex. ett $(2, 4, 8)$-spel se ut i Vattenfallsnotation?

13. Kan ni använda detta för att lösa spelet från den 3e augusti?

14. **Cirkelspel.** Det här spelet spelas med ett papper med en cirkel på och några markörer (alla markörer måste vara likadana, men annars spelar det ingen roll hur de ser ut, så länge de får plats i cirkeln). De två spelarna turas om att göra varsitt drag. Ett drag går ut på att lägg en av markörerna i cirkeln. Markörerna får inte nudda cirkelns kant, och de får aldrig ligga på varandra. Man får inte heller flytta redan lagda markörer. Den första spelaren som inte kan hitta en plats att lägga en markör på förlorar.

En spelare har en vinnande strategi - vilken? Vad är strategin?

10. Cykliska fyrhörningar

7 augusti

För 3 godtyckliga punkter som inte ligger på en linje finns det en cirkel som går igenom alla 3 av dessa. Detta är dock inte sant för 4 godtyckliga punkter (tänk till exempel på ett parallellogram). Om 4 punkter A, B, C och D faktiskt ligger på en cirkel kallas fyrhörningen $ABCD$ för en cyklisk fyrhörning. Från randvinkelsatsen är det uppenbart att om $ABCD$ är cyklisk så är $\angle ACB = \angle ADB$. Det är ingen överraskning denna implikation även gäller åt det andra hållet.

1. Låt C och D ligga på samma sida av linjen AB så att $\angle ACB = \angle ADB$. Visa att fyrhörningen $ABCD$ är cyklisk.

2. Låt B och D ligga på olika sidor av linjen AC så att $\angle ABC + \angle ADC = 180^\circ$. Visa att $ABCD$ är cyklisk.

Cykliska fyrhörningar är det starkaste redskapet inom vinkeljakt, de förekommer inom de allra flesta tävlingsproblem. En mycket rimlig problemlösnings strategi är att rita en bra bild (med passare och linjal) och sedan kolla vilka fyrhörningar som ser cykliska ut. Cykliska fyrhörningar är så användbara tack vare hur mycket information de innehåller. Om vi lyckas bevisa att $\angle ACB = \angle ADB$ följer alla följande påståenden.

- $\angle ABC = 180^\circ - \angle ADC$,
- $\angle ABD = \angle ACD$,
- $\angle BAC = \angle BDC$,
- $\angle BAD = 180^\circ - \angle BCD$,
- $\angle CAD = \angle CBD$.

Så från att ha bevisat ett påstående följer 5 nya!

3. Räkna ut vinkeln $\angle DAB$ från uppgift 2 i vinkeljakts uppgifterna.

4. Låt $ABCD$ vara en symmetrisk parallelltrapets där AB och CD är parallella och $AD = BC$. Visa att $ABCD$ är cyklisk.

5. Låt AD och BE vara två av höjderna i triangeln ABC. Visa att $\angle ABE = \angle ADE$.

6. Triangeln ABC är inskriven i cirkeln ω. Linjen l är parallell med tangenten vid A och skär AB och AC vid punkterna D och E. Visa att $BCED$ är cyklisk.

7. Låt ω_1 och ω_2 vara två cirklar utan skärningspunkter. Kordan AB av ω_1 och CD av ω_2 är parallella. AC skär ω_1 vid E och BD skär ω_1 vid F. Visa att $CDFE$ är cyklisk.

8. Triangeln ABC har en rät vinkel vid A. En cirkel med mittpunkt på sidan AB som går igenom punkten B skär AB och BC vid D och E. Visa att $\angle AED = \angle ACD$.

9. Cirklarna ω_1 och ω_2 skär varandra i punkterna A och B. En linje genom A skär ω_1 och ω_2 i C och D. En linje genom B skär ω_1 och ω_2 i E och F. Visa att $CDFE$ är en parallelltrapets.

10. I triangeln ABC är AD en av höjderna. Låt E och F ligga på AB och AC så att $DE \perp AB$ och $DF \perp AC$. Visa att $\angle FBE = \angle FCE$.

11. Vända brickor III

8 augusti

1. Lägg några spelbrickor på bordet; en del med svart sida upp och en del med vit sida upp. Med brickorna spelar två spelare ett spel mot varandra. I ett drag tas två brickor bort från bordet. Om två likadana brickor togs bort måste en svart bricka läggas tillbaka. Om två brickor av olika färg tas bort måste en vit bricka läggas tillbaka.

   Spelet spelas till dess att bara en bricka finns kvar. Spelare ett vinner om denna bricka är svart, spelare två vinner om den är vit. Vem vinner spelet om båda spelarna spelar perfekt?

2. Två spelare turas om att lägga spelpjäser i hörnen på en regelbunden 9-hörning. I ett drag får spelaren själv välja vilken färg brickan ska ha upp. Spelet är slut när alla hörnen har fått en bricka på sig.

   Spelare ett vinner om det finns brickor av samma färg som ligger i hörnen av en likbent triangel. Annars vinner spelare två. Vem vinner spelet om båda spelar perfekt?

3. Två spelare turas om att lägga brickor i en tom 4×4-kvadrat. I ett drag väljer spelaren själv om hen ska lägga ner brickan med svart eller vit sida upp.

   Den som i ett drag först skapar en enfärgad delkvadrat av storlek 2×2 förlorar. Vem vinner spelet?

4. Två spelare turas om att lägga spelbrickor i en tom 3×9-rektangel. I ett drag väljer spelaren själv om hen vill lägga brickan med svart eller vit sida upp.

   Spelare ett vinner om hen kan undvika att det skapas en delrektangel vars hörn har samma färg. Annars vinner spelare två. Vem vinner vid perfekt spel?

5. Två spelare turas om att lägga spelbrickor i en tom 3×7-rektangel. I ett drag väljer spelaren själv om hen vill lägga brickan med svart eller vit sida upp.

Spelare ett vinner om hen kan undvika att det skapas en delrektangel vars hörn har samma färg. Annars vinner spelare två. Vem vinner vid perfekt spel?

Extraproblem

6. Två spelare lägger spelbrickor i en tom jättestor 15×15-kvadrat. I ett drag lägger en spelare en bricka med en av tre tillgängliga färger (säg svart, vit och röd) uppåt. Spelaren väljer själv vilken färg hen vill.

Spelare ett vinner om det någon gång skapas två kolumner som innehåller lika många brickor av någon av färgerna.

12. Cykliska fyrhörningar II

8 augusti

1. Låt $\omega_{1,2,3,4}$ vara fyra cirklar så att ω_1 och ω_3 tangerar ω_2 och ω_4. Visa att de fyra tangeringspunkterna ligger på en cirkel.

2. Låt $\omega_{1,2,3}$ vara tre cirklar och l en linje så att ω_1 och ω_3 tangerar ω_2 och l. Visa att de fyra tangeringspunkterna ligger på en cirkel.

Dessa två uppgifter är väldigt lika varandra, nästan så pass att vi skulle vilja kalla de för samma uppgift. Detta är faktiskt inte helt fel, vi kan låta radien av cirkeln ω_4 gå mot oändligheten, i detta fall går ω_4 mot l och de fyra tangeringspunkterna förblir cykliska. Att betrakta en linje som en cirkel med en väldigt stor radie kan ibland vara användbart, tyvärr är en djupare diskussion om detta tema överkurs, men jag kan djupt rekommendera alla intresserade att läsa något om geometrisk inversion där liknande idéer tas upp.

3. Låt A, B, C, D, X, Y, Z, W vara 8 punkter i planet så att $ABYX$, $BCZY$, $CDWZ$, $DAXW$ och $XYZW$ alla är cykliska. Visa att $ABCD$ är cyklisk.

4. Låt D, E och F vara punkter på sidorna BC, CA och AB i triangeln ABC. Visa att de tre cirklarna som går igenom AEF, BFD och CDE har en gemensam punkt.

5. Låt O ligga i origo och A och B ligga på den positiva y-axeln. Låt P vara en punkt på cirkeln med diameter AB. Linjen PA skär x-axeln vid Q. Visa att $\angle PQB = \angle POB$.

6. E och F ligger på sidan BC av fyrhörningen $ABCD$. Vi är givna $\angle BAE = \angle CDF$ och $\angle EAF = \angle FDE$. Visa $\angle FAC = \angle EDB$.

7. Låt O ligga innanför parallellogrammet $ABCD$ så att $\angle AOB + \angle COD = 180°$. Visa att $\angle OBC = \angle ODC$.

13. Vända brickor IIII

9 augusti

1. En magiker lägger fram några trolleribrickor på ett bord. Trolleribrickorna har två identiska sidor, bortsett från att en sida är svart och att en är vit.

Magikern tar på sig en ögonbindel och ber någon ur publiken att vända och flytta runt på brickorna hur mycket hen vill. När åskådaren är färdig ber magikern om att få veta hur många brickor som ligger med vit sida upp.

Efter att åskådaren berättat hur många som låg med vit sida upp, tar magikern, fortfarande med ögonbindel på, och delar brickorna i två höger med lika många vita pjäser i båda. Hur gjorde magikern för att lyckas med tricket?

2. Nu ber magikern en åskådare att lägga ut fyra trolleribrickor på rad på bordet, så att det finns åtminstone en bricka av varje färg.

Magikern förklarar att hen ska ordna så att alla brickor ligger med samma sida upp. Vidare säger hen att det enda hen får göra är att ställa frågan 'ligger alla brickor med samma sida upp?', och att hen får vända på brickorna hur mycket hen vill. Magikern har lyckats när den ärliga åskådaren svarar 'ja'.

Hur kan magikern göra för att (garanterat) lyckas med tricket? Om magikern är riktigt skarp, hur många gånger behöver hen fråga om brickorna ligger med samma sida upp?

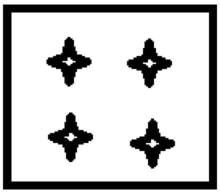
3. Hur kan magikern göra om hen istället ber åskådaren att från början lägga ut fem brickor?

4. Magikern ber nu en åskådare att lägga ut sex brickor på bordet, så att åtminstone en bricka med varje färg förekommer. Magikern förklarar att hen ska ordna så att det till slut finns tre vita och tre svarta brickor, och att det enda hen får göra är att vända brickor samt fråga 'är det tre vita och tre svarta på bordet?'.

Om åskådaren svarar 'nej' vänder magikern på vissa av brickorna och frågar igen. Visa att magikern kan lyckas med tricket. Hur många frågor behöver magikern ställa?

5. Magikern har nu placerat ett trolleribräde i form av ett papper med fyra små

markeringar på sitt bord. Dessa markeringar befinner sig i hörnen på en kvadrat, och ska snart blir täckta med trolleribrickor.



Ett trolleribräde.

Magikern ber någon av åskådarna att komma fram till bordet och placera ut en bricka per markering, så att det åtminstone finns en bricka av varje färg. Magikern förklarar att det trick hen ska utföra slutar med att alla brickor ligger med samma färg upp. Vidare berättar magikern att det enda hen får göra är att vända på brickor, vilka hen vill, och efter hen vänt färdigt fråga åskådaren 'ligger alla brickor med samma färg upp?'. Om det är sant ska åskådaren svara 'ja' och publiken ska klappa händerna.

Om det är falskt ska åskådaren svara 'nej'. Magikern berättar till sist att efter ett 'nej'-svar får åskådaren rotera pappret med spelpjäserna ett fjärdedels varv, två fjärdedels vars eller tre fjärdedels varv om hen så önskar. Efter en eventuell rotation av pappret vänder magikern på några brickor och frågar om 'alla brickor är med samma sida upp?'.

Frågor att attackera i någon ordning:

(a) hur kan magikern göra för att (garanterat) lyckas med tricket?

(b) hur många gånger behöver magikern ställa frågan 'ligger alla brickor med samma sida upp?' som mest?

(c) säg att åskådaren vill att magikern ska ordna så att alla brickor ligger med svart sida upp, och inte bara att alla brickor ligger med samma sida upp. Kan magikern klara av detta?

(d) om ja, hur många gånger behöver magikern fråga som mest?

14. Triangelns mittpunkter

9 augusti

En cirkel har en entydig mittpunkt, men när det gäller trianglar kan mittpunkten definieras på olika sätt. En fysiker skulle antagligen kalla tyngdpunkten för mittpunkten, men det finns många andra val man kan göra. Encyklopedia of Triangle Centers har en lista på 32784 olika punkter som på ett sätt eller annat kan kallas för triangelns mittpunkt. Vi kommer endast att ta upp de tre viktigaste ur ett problemlösnings perspektiv, omcentrum, orthocentrum och incentrum.

Omcentrum

1. Låt ABC vara en triangel. Visa att mittpunktsnormalerna av de tre sidorna skär varandra i en gemensam punkt.

Denna gemensamma punkt heter omcentrum, betecknas oftast med O och är mittpunkten av omcirkeln, cirkeln som triangelns hörn ligger på. Detta visar även att för tre godtyckliga punkter som inte ligger på en linje så finns det en cirkel som går igenom alla tre.

2. Vinkeljaga omcentrum. Räkna ut vinklarna $\angle AOB$ och $\angle ABO$ givet att $\angle A = \alpha$ och $\angle B = \beta$.

Orthocentrum

3. Låt ABC vara en triangel. Visa att de tre höjderna skär varandra i en gemensam punkt.

Denna punkt kallas orthocentrum och betecknas oftast med H.

4. Vinkeljaga omcentrum. Låt höjdernas fotpunkter från A, B och C vara D, E och F. Räkna ut vinklarna $\angle AHB$, $\angle DHE$ och $\angle DEA$ givet att $\angle A = \alpha$ och $\angle B = \beta$.

Incentrum

5. Låt ABC vara en triangel. Visa att de tre bisektriserna skär varandra i en gemensam punkt.

Denna punkt kallas incentrum och betecknas ofta med I. Den är mittpunkten av incirklen, cirkeln som tangerar triangelns tre sidor (varför?). Den är relativt jobbig att arbeta med och kräver oftast ”längdjakt” istället för vinkeljakt samt fulare metoder som trigonometri.

6. Vinkeljaga incentrum. Räkna ut $\angle AIB$ givet att $\angle A = \alpha$ och $\angle B = \beta$.

Uppgifter

I alla uppgifter jobbar vi med triangeln ABC med omcentrum O, orthocentrum H och incentrum I om inget annat anges.

7. Låt AI skära omcirkeln vid punkten D. Visa att $DI = DC = DB$.

Den föregående uppgiften kallas superlemma och är användbar tack vare att den kopplar ihop incentrum med omcirkeln.

8. Låt O' och H' vara spegelbilderna av O och H på linjen AB. Visa att O' och H' ligger på omcirkeln.

9. Visa att spegelbilden av linjen AH på linjen AI är linjen AO. (Tips: Visa $\angle BAH = \angle CAO$.)

Den föregående uppgiften är så fin att man knappt tror att det är sant.

10. Låt höjdernas fotpunkter vara D, E och F. Visa att H är incentrum i triangeln DEF.

11. I en cyklisk fyrhörning $ABCD$ är I_1 och I_2 incentrum i trianglerna ABC och DBC. Visa att $I_1 I_2 BC$ är cyklisk.

12. Låt X, Y och Z vara mittpunkterna av bågarna AB, BC, och CA av omcirkeln. Visa att I är orthocentrum i triangeln XYZ.

13. Låt höjdernas fotpunkter från A, B och C vara D, E och F. Låt EF skära omcirkeln i punkten P. Linjen BP och DF skär varandra i punkten Q. Visa att $AP = AQ$.

Den sista uppgiften är från Internationella Matematikolympiaden, men kan lösas med hjälp av cykliska fyrhörningar. Ett tips: rita en stor och punktlig bild (med passare och linjal) och kolla om någon av fyrhörningarna verkar vara cykliska. Vad sägs om $AQPF$?

15. Mattedrabbning blå

10 augusti

Lag 1 - Lag 2

1. Fisken (inklusive rens) kostar 200 kr/kilot, fiskfilén utav samma fisk kostar 230 kr/kilot, medan fiskrens kostar 80 kr/kilot. Hur många gram rens är det i ett kilo fisk?

2. Två spelare spelar ett spel på ett 4×4-bräde. Spelare ett börjar med att färga en ruta blå, vilken hen vill. Därefter ställer spelare två en blå spelpjäs i någon annan ruta. Spelare ett ska nu gå med den blå spelpjäsen från ruta till ruta och på så sätt besöka alla brädets rutor precis en gång, och till sist sluta i den blå rutan. I ett drag är det endast tillåtet för spelare ett att flytta spelpjäsen ett steg åt höger, ett steg åt vänster, ett steg uppåt och ett steg neråt.

Spelare ett vinner spelet om hen lyckas, annars vinner spelare två. Vem vinner?

3. En get och en kossa ska springa fram och tillbaka över en stor avlång äng. Geten står vid en av den rektangulära ängens kortsidor, mitt emot på andra kortsidan står kossan. De börjar springa mot varandra samtidigt och springer båda med konstanta hastigheter samt parallellt med en av ängens långsidor. Efter en kort stund passerar de varandra ute på ängen, 250 meter bort från närmsta kortsida. När djuren sprungit över till respektive sida, stannar de och vilar tre minuter innan de börjar springer tillbaka. På tillbakavägen passerar de varandra 100 meter bort från den andra kortsidan.

Hur lång är ängen?

4. Bestäm alla par av heltal (x, y) som löser ekvationen eller bevisa att inga lösningar finns:

$$2x^2 - 1 = 5y.$$

5. Vi startar på en punkt med koordinater $(1, 0, 0)$ i rummet. Ett 3-dimensionellt springardrag ändrar en koordinat med 1, en annan koordinat med 2 och den kvarvarande koordinaten med 3. Exempelvis så kan vi ta oss från $(2, 2, 2)$ till $(3, -1, 0)$. Kan vi ta oss från $(1, 0, 0)$ till $(0, 0, 0)$ med en följd av sådana springardrag?

6. Två bisektriser i en triangel skär varandra med vinkeln $110°$. Bestäm triangelns tredje vinkel.

Lag 3 - Lag 4

1. I triangeln ABC är $AB = AC$. Vi ritar en cirkel genom B och C som skär AB och AC i punkterna D respektive E. Givet att $BE = BC$ och $CE = DE$, räkna ut triangelns vinklar.

2. På bordet ligger 125 tändstickor. På ett drag får man ta bort 1, 2, 3, 4 eller 5 tändstickor från högen. Den spelare som inte kan göra ett drag (när tändstickorna tagit slut) förlorar spelet. Vem vinner om båda spelar optimalt, första eller andra spelaren?

3. En korg innehåller minst tre gula bollar och några röda bollar. Om tre bollar slumpmässigt skulle plockas upp ur korgen skulle sannolikheten att få upp tre gula vara p. Om korgen istället hade haft ytterligare en gul boll i sig, skulle sannolikheten varit $4p/3$. Hur många röda bollar kan som mest finnas i korgen?

4. Linjen l tangeras av cirkeln ω_1 vid punkten A och av cirkeln ω_2 vid punkten B. De två cirklarna skär varandra i punkterna C och D. Visa att $\angle ACB + \angle ADB = 180°$.

5. Låt p_1 och p_2 vara två på varandra följande primtal större än 2. Visa att deras summa kan skrivas som en produkt av tre heltal som alla är större än 1 (behöver inte vara olika tal).

6. En infektion sprids på ett 8×8 schackbräde. I början är 7 rutor infekterade. En ruta blir infekterad om minst två av dess grannar är infekterade. Kan infektionen nå alla 64 rutor?

Lektionsmaterial lila gruppen

16. To infinity and beyond!

2 augusti

1. Låt oss betrakta två mängder $X = \{a, b, c, d, e, f, g, h\}$ och $Y = \{A, B, C, D, E, F, G, H\}$. Valentina påstår att mängderna är lika stora (dvs innehåller lika många element), men kan inte förklara varför för Johan eftersom han inte kan räkna med så stora tal. Kan du förklara varför mängderna har lika många element utan att räkna dem?

2. Kan du ge en definition på när två mängder har lika många element, (utan att involvera att verkligen räkna elementen)?

3. Testa din definition!

(a) Visa att varje mängd är lika stor som sig själv.

(b) Visa att om mängden A är lika stor som mängden B och om mängden B är lika stor som mängden C så är mängden A lika stor som C.

Vi säger alltså att två mängder X, Y är lika stora om vi kan hitta en funktion f från X till Y så att $f(x_1) = f(x_2)$ innebär att $x_1 = x_2$ (två olika saker måste hamna på olika ställen) samt att för varje $y \in Y$ så existerar det ett $x \in X$ så att $f(x) = y$ (dvs vi "träffar" varje grej i y). Mindre formellt så innebär det att vi kan para ihop elementen från X och Y så att alla får precis en "kompis".

4. Visa att det finns lika många positiva heltal som det finns negativa heltal.

5. Visa att mängden av ickenegativa heltal $\{0, 1, 2, 3, ...\}$ är lika stor som mängden av positiva heltal $\{1, 2, 3, 4, ...\}$

6. Visa att det finns lika många heltal som det finns jämna heltal.

7. Visa att följande par av mängder är lika stora:

(a) Positiva heltal och heltal.

(b) Reella tal x sådana att $1 < x < 2$, och reella tal y sådana att $5 < y < 6$.

(c) Reella tal x sådana att $0 < x < 1$, och reella tal y sådana att $1 < y < 10$.

(d) Positiva reella tal x och reella tal y sådana att $0 < y < 1$.

(e) Reella tal och reella tal utom nollan.

(f) Heltal och par av heltal. (mängder som är lika stora som heltalen kallas uppräkneliga)

(g) Heltal och rationella tal (dvs bråktal).

(h) Reella tal och par av reella tal (dvs är planet lika stort som linjen)?

8. En talföljd är en sekvens av tal (kan vara heltal, reella tal etc beroende på situationen, i denna uppgift så antar vi att det är heltal). Ett exempel kan vara följden $1, 2, 3, 4, \ldots$. Visa att samtliga talföljdsmängder nedan är uppräkneliga.

(a) Mängden av talföljder som efter ett tag blir 0.

(b) Mängden av talföljder som efter ett tag blir konstant (exempelvis $1, 2, 3, 3, 3, 3, \ldots$).

(c) Mängden av talföljder som efter ett tag blir periodiskt (dvs den börjar upprepa sig, t.ex. $1, 2, 3, 4, 3, 4, 3, 4, \ldots$).

9. Visa att mängden av alla talföljder (av heltal) INTE är lika stor som mängden av alla heltal. Tips: Antag motsatsen, då har vi en hopparning med positiva heltalen, dvs vi har talföljd nr 1, talföljd nr 2 osv. Kan du garanterat hitta en talföljd som inte finns med på vår lista?

10. Använd idén från föregående uppgift för att visa att det inte finns lika många heltal som reella tal.

11. Algebraiska tal är tal som är lösningar till polynomekvationer i en variabel med heltalskoefficienter (exempelvis så är $\frac{1}{\sqrt{2}}$ algebraiskt då det är en lösning till $2x^2 - 1 = 0$. Visa att de algebraiska talen är uppräkneliga. Hur stor är mängden av icke-algebraiska reella tal (dvs de reella tal som INTE är lösning till polynom med heltalskoefficienter)?

12. Låt A vara en samling av disjunkta intervall på formen $a < x < b$. Kan vi hitta överuppräkneligt många sådana intervall?

13. Är mängden av alla ändliga delmängder till $\mathbb{Z}$ uppräknelig eller överuppräknelig?

14. Hur skulle vi kunna ge definitioner på att X är högst lika stor som Y ($|X| \leq |Y|$) och minst lika stor som Y ($|X| \geq |Y|$)?

Finns det större oändligheter än de reella talen? Möjligt projekt...

17. Polynom I

2 augusti

Definition 1. Ett reellt (komplext) polynom i variabeln x är ett uttryck som kan skrivas som $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_2x^2 + a_1x + a_0$, där de olika talen a_i är reella (komplexa) tal. (Notera att det inte måste vara just x som är variabeln - det går egentligen lika bra med vilket tecken som helst.)

Definition 2. *Graden* på $p(x)$ betecknas $\deg p(x)$ (från engelskans «degree») och är lika med det största i så att $a_i \neq 0$; om $a_i = 0$ för alla i så säger vi att graden är $-\infty$.

1. Vilka av följande är polynom? Om det är ett polynom, ange dess grad.

(a) 1

(b) $2^x x^2 + 1$

(c) $0 \cdot x^{100} + 3x^4 - 5$

2. Polynomet $p(x)$ har icke-negativa heltal som koefficienter. Ni vet att $p(1) = 4$ och $p(5) = 136$. Bestäm alla polynomets koefficienter.

Definition 3. Vi säger att ett polynom $q(x)$ delar $p(x)$ om det finns ett polynom $r(x)$ så att $p(x) = q(x) \cdot r(x)$. Detta betecknas $q(x)|p(x)$.

3. Bevisa att om $q(x)$ delar $p(x)$ och $q(a) = 0$ så är också $p(a) = 0$.

4. Att dela polynom liknar att dela heltal - ibland går det inte jämnt ut. Visa att $x+1$ inte delar x^2+1, dvs. att det inte finns något polynom $r(x)$ så att $r(x) \cdot (x+1) = x^2 + 1$.

Definition 4. Om polynomet $a(x)$ kan skrivas som $a(x) = q(x)b(x) + r(x)$, där $\deg r < \deg b$, så säger vi att $q(x)$ är *kvoten* av $\frac{a(x)}{b(x)}$, och kallar $r(x)$ *resten*.

5. Bevisa att för alla polynom $a(x)$ och $b(x) \neq 0$ finns det precis ett sätt att skriva $\frac{a(x)}{b(x)}$ på den här formen, så att $q(x)$ och $r(x)$ entydigt bestämda.

6. Bevisa att om $p(a) = 0$ så gäller $(x-a)\,|\,p(x)$. Jämför med uppgift 4 ovan.

7. Antag att polynomet $p(x)$ har rest -1 vid division med $x+2$ och rest 2 vid division med $x-1$. Vad har det för rest vid division med $(x+2)(x-1)$?

8. Antag att $p(x)$ bara har heltalskoefficienter, och att inget av de 3 heltalen $p(8)$, $p(9)$ och $p(10)$ är delbart med 3. Visa att i så fall finns det inget heltal a så att $x-a$ delar $p(x)$.

För att enkelt för hand dela två polynom finns det en ganska enkel process, kallad *polynomdivision*. Den går ut på att "lösa ut" en exponent i taget, med början i de högsta.

■ **Exempel 1** Dela $t(x) = x^4 + 2x^3 - 4x^2 + 7x - 5$ med $n(x) = 3x^2 - x + 5$.

Börja med att räkna ut vad första termen i $t(x)$ delat med första termen i $n(x)$ blir - i det här fallet får vi $\frac{1}{3}x^2$; vi kallar detta $q_1(x)$. Detta blir då första termen i kvoten. Räkna nu ut $t(x) - q_1(x)n(x) = \frac{7}{3}x^3 - \frac{17}{3}x^2 + 7x - 5$; kalla detta $t_1(x)$.

I nästa steg så gör vi samma sak igen, men ersätter $t(x)$ med $t_1(x)$ - dvs. definiera $q_2(x)$ som första termen i $t_1(x)$ delat med första termen i $n(x)$, alltså $q_2(x) = \frac{7}{9}x$. Beräkna $t_2(x) = t_1(x) - q_2(x)n(x) = \frac{44}{9}x^2 + \frac{28}{9}x - 5$.

I sista steget får vi $q_3(x) = -\frac{44}{27}$, $t_3(x) = t_2(x) - q_3(x)n(x) = \frac{40}{27}x + \frac{85}{27}$. Notera att $\deg t_3(x) < \deg n(x)$, alltså skulle $q_4(x)$ inte bli ett polynom - det skulle ha x i nämnaren. Därmed är vi nu klara, och **kvoten** blir $q_1(x) + q_2(x) + q_3(x) = \frac{1}{3}x^2 + \frac{7}{9}x - \frac{44}{27}$ och **resten** blir $t_3 = \frac{40}{27}x + \frac{85}{27}$. ■

9. Förenkla $\frac{81x^4+9x^2+1}{9x^2+3x+1}$.

10. Anta att a, b, c är reella tal, $p(x) = x^2 + \frac{1}{2}x + a$, $q(x) = x^2 + bx + c$ och att $p(x)q(x) = q(p(x))$ för alla x. $p(q(x))$ har ett nollställe i $x = \frac{1}{2}$. Hitta alla andra reella nollställen till $p(q(x))$.

11. Bevisa att polynomet $x^6 - x^5 + x^4 - x^3 + x^2 - x + \frac{3}{4}$ inte har några reella nollställen.

Extrauppgifter

12. $p(x)$ är ett polynom med heltalskoefficienter. Det finns två olika heltal a och b så att $p(a)p(b) = -(a-b)^2$. Visa att $p(a) + p(b) = 0$.

13. Polynomet $p(x)$ har heltalskoefficienter, och ni vet att $p(5) = 101$ och $p(18) = 57$. Visa att $p(x)$ inte har någon heltalsrot.

14. För ett polynom p vet ni att $p(0) = 1$ och att $(p(x))^2 = 1 + x + x^{100}q(x)$ för något polynom $q(x)$. Visa att koefficienten framför x^{99} i polynomet $(p(x)+1)^{100}$ är lika med noll.

15. Av de tre reella polynomen p, q, r är minst en ett tredjegradspolynom och minst en ett andragradspolynom. Ni vet att $(p(x))^2 + (q(x))^2 = (r(x))^2$. Visa att minst en av p, q, r är ett tredjegradspolynom med tre reella nollställen.

18. Avstånd

3 augusti

1. Avståndet mellan två punkter i planet anges med en funktion. Hur skrivs den? Vilka mängder är involverade?

Definition 1. Avståndet mellan två punkter i $\mathbb{R}^n$ anges med

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2}$$

där punkten x har koordinaterna $(x_1, x_2, \ldots, x_n)$ (och motsvarande för y).

2. Hur lång pinne får plats i en kub? Vad händer om dimensionen blir stor? Kan man få plats med en pinne med längden 10 meter i en (n-dimensionell hyper-) kub med sidan 1 mm? Vilken dimension krävs?

3. Studera en sfär med radie 1 och en kub med sidlängd 1. Rita gärna en bild. Visa att kuben går att placera i sfären utan att någon del sticker ut. Går samma sak i höga dimensioner? Vad händer om vi har en sfär med radie 10 meter och en kub med sidlängd 1 millimeter?

4. Tag en kvadrat med sida 2 i planet. Vi kan lätt skriva in 4 cirklar med diameter 1 så att de precis rör vid varandra och vid kvadratens kanter. I mitten så placerar vi en liten cirkel, som precis snuddar de andra cirklarna. Vad är dess radie? Vad händer i högre dimensioner (fundera först på hur problemet ska formuleras, hur många sfärer behövs)?

5. En mängd med en avståndsfunktion $d(x,y)$ kallas ett metriskt rum (och funktionen $d(x,y)$ kallas då en metrik, utläses som $d(x,y) =$ avståndet från x till y). I denna problem så låter vi vår mängd bestå av tre punkter, $0, 1$ och 2. Alla avstånd är inte alltid utsatta. Nedan är några exempel på förslag till avståndsfunktioner, varför är de INTE är rimliga metriker (från ett intuitivt perspektiv)? Skriv upp vilken/vilka "regler" som du tycker metriker borde ha som den bryter mot. Vi låter sedan dessa regler bilda definitionen av en metrik.

(a) $d(0,0) = 0, d(1,1) = 0, d(1,0) = 1, d(0,1) = 2$

(b) $d(0,x) = 1$

(c) $d(0,1) = 1, d(0,2) = 0, d(1,2) = 1$

(d) $d(0,1) = 3, d(2,1) = 10, d(2,0) = 4$

6. Vi säger att en cirkel $C(a,r)$ centrerad i a och med radie r består av alla punkter x så att $d(a,x) = r$. Vi säger att en boll $B(a,r)$ centrerad i a och med radie r består av alla punkter x så att $d(a,x) < r$. Hur skulle man kunna tänka sig generalisera ett linjesegment mellan punkterna a och b (detta är inte självklart och det finns olika sätt att generalisera på beroende på vilken egenskap man vill fånga, så fundera på något!)? Sitt inte för länge med problemen, men fundera på vilka egenskaper man skulle kunna vilja fånga.

7. Vi har tidigare tittat på den vanliga metriken på $\mathbb{R}^2$. I denna uppgift så ska vi studera den så kallade Manhattan-metriken. Den kommer från Manhattans gatstruktur där det går gator nästan överallt, men de går bara nord-syd eller öst-väst. Det innebär att ska man ta sig mellan två punkter, så kan man inte använda diagonaler/hypotenusor, utan man får bara gå rakt nord/syd eller öst/väst.

- **(a)** Ge en formel för avståndet mellan två punkter (x_1, x_2) och (y_1, y_2).
- **(b)** Visa att det vi har faktiskt är en metrik, med din definition ovan.
- **(c)** Hur ser cirklar och bollar ut i Manhattanmetriken?
- **(d)** Man kan definiera en ellips som alla punkter p så att $d(a,p) + d(b,p) = r$ för något värde på r och givet punkter a, b (som kallas ellipsens brännpunkter). Hur ser en ellips ut på Manhattan?

8. En något tråkigare metrik är den diskreta metriken. Den ges av att $d(x,y) = 0$ om $x = y$ och $d(x,y) = 1$ annars. Hur ser cirklar och ellipser ut i denna metrik? Grundmängden kan väljas som t.ex. $\mathbb{R}^2$.

9. Det är rätt nära till hands att fundera på begrepp som längd, area och volym när man väl pratat om avstånd. Låt oss kika på mängder på linjen (med vanliga metriken etc). Vi ska i denna problem titta på en intressant mängd, Cantormängden. Vi använder notationen $[a,b]$ för det slutna intervallet $a \leq x \leq b$.

- **(a)** Vi startar med det slutna intervallet $[0,1]$. Sedan så tar vi bort den mittersta tredjedelen, och får kvar två mindre intervall, $[0, \frac{1}{3}]$ och $[\frac{2}{3}, 1]$. Vi tar sedan bort den mittersta tredjedelen på var och en av dom och så vidare. Eventuella punkter som stannar kvar kallar vi för C, Cantormängden. Vilken längd bör C ha totalt?
- **(b)** Visa att C innehåller någon punkt.
- **(c)** Visa att C är lika stor som intervallet vi startade med (i mängdmening). (rätt lurig, fråga gärna om tips)
- **(d)** Visa att mellan varje par av Cantortal så finns det något reellt tal som inte är ett Cantortal.
- **(e)** Kan du genom att ta bort intervall av olika storlekar hitta en variant av Cantormängden som har längd $\frac{1}{2}$?

10. En metrik som används rätt ofta i talteoretiska sammanhang är p-adiska tal metriken på de rationella talen. Där väljer vi först ett primtal p. Sedan så säger vi att $|p^a \frac{r}{s}| = p^{-a}$ (där r och s saknar gemensam delare och inte heller delas av p, och $|0| = 0$).

(a) Visa att $d(x,y) = |x-y|$ är en metrik.

(b) När är tal "nära" varandra i p-adisk metrik?

(c) Möjligt projekt för senare:Visa att en kvadrat inte kan delas upp i ett udda antal trianglar med samma area (rätt lurigt, men enda beviset jag känner till använder p-adisk norm och lite andra grejer).

11. En metrik kallas en ultrametrik om den uppfyller $d(x,y) \leq max(d(x,z), d(z,y))$. Vilka av tidigare metriker är ultrametriker?

12. Ultrametriker har flera intressanta egenskaper. Visa följande:

(a) Om två bollar har ett gemensamt element så ligger den ena inuti den andra.

(b) Varje punkt inuti en boll är dess mittpunkt.

(c) Försök hitta på ett eget exempel på ultrametrik.

13. Hur skulle man kunna definiera avstånd mellan delmängder i $\mathbb{R}^2$? Ni kan anta att delmängderna är snälla och har trevliga egenskaper (så fokusera mer på att hitta en idé än att se att allt är väldefinierat etc). Vi skulle gärna vilja ha en metrik som är hyfsat intuitiv, t.ex. att avståndet mellan två delmängder som innehåller precis en punkt var är det vanliga avståndet.

19. Affina avbildningar I

3 augusti

Definition 1. En avbildning f av planet på sig självt kallas för *affint* om den avbildar lika vektorer på lika, och det dessutom gäller att $f(\overrightarrow{a} + \overrightarrow{b}) = f(\overrightarrow{a}) + f(\overrightarrow{b})$ och $f(k\overrightarrow{a}) = kf(\overrightarrow{a})$. Dessutom gäller att endast nollvektorn avbildas på nollvektorn.

Definition 2. En avbildning f av planet på sig självt kallas för *affint* om det existerar ett par av koordinatsystem $O\overrightarrow{a}\,\overrightarrow{b}$ och $O'\overrightarrow{a}'\,\overrightarrow{b}'$ så att en punkts koordinater i $O\overrightarrow{a}\,\overrightarrow{b}$ är desamma som koordinaterna i $O'\overrightarrow{a}'\,\overrightarrow{b}'$ utav punktens bild, dvs om $\overrightarrow{OC} = x\overrightarrow{a} + y\overrightarrow{b}$ så är $\overrightarrow{O'C'} = x\overrightarrow{a}' + y\overrightarrow{b}'$ för alla punkter C.

Egenskaper hos affina avbildningar:

- Den inversa avbildning till en affin avbildning kommer också vara affin.
- Räta linjer avbildas på räta linjer.
- Parallella linjer avbildas på parallella; linjer som skär varandra avbildas på linjer som skär varandra.
- Förhållandet mellan sträckor som ligger på samma eller parallella linjer bevaras.
- För två godtyckliga trianglar ABC och $A'B'C'$ finns det exakt en affin avbildning som avbildar ena triangeln på den andra så att A avbildas på A', B på B' och C på C'.

1. Bevisa att de två definitionerna är ekvivalenta.

2. Visa att:

(a) Varje parallellogram är affint ekvivalent med en kvadrat.

(b) Varje parallelltrapets är affint ekvivalent med någon likbent trapets (men inte alla likbenta).

(c) Två konvexa fyrhörningar är affint ekvivalenta om och endast om diagonalernas skärningspunkt delar diagonalerna i samma respektive förhållande.

3. Visa med hjälp av affina avbildningar att för varje parallelltrapets så ligger basernas mittpunkter, diagonalernas skärningspunkt samt sidolinjernas skärningspunkt på en och samma linje.

4. I parallelltrapetsen $ABCD$ med baserna AD och BC har man dragit en linje genom B som är parallell med sidan CD och som skär diagonalen AC i punkten P. Genom punkten C har man dragit en linje som är parallell med sidan AB och skär diagonalen BD i punkten Q. Visa att linjen PQ är parallell med trapetsens baser.

5. I en spetsvinklig triangel ABC har man dragit höjden AH, medianen AM, medianen BN och cevianen BG, så att punkten G delar sidan AC i samma förhållande som punkten H delar sidan BC. Låt BN och AH skära varandra i punkten K, och AM och BG i punkten L. Visa att KL är parallell med AB.

6. Genom varje hörn på en triangel har man dragit två linjer som delar den motstående sidan i tre lika stora delar. Visa att diagonalerna som förbinder motsatta hörn av sexhörningen som bildas av dessa linjer skär varandra i samma punkt.

7. Ett parallellogram $ABCD$ är given. En godtycklig linje skär strålarna AB, AC och AD i punkterna P, Q och R respektive. Visa att $\dfrac{AB}{AP} + \dfrac{AD}{AR} = \dfrac{AC}{AQ}$.

8. Varje diagonal i en konvex femhörning är parallell med någon utav dess sidor. Visa att det finns en affin avbildning som avbildar denna femhörning på en regelbunden femhörning.

9. En triangel ABC är given. Låt O vara medianernas skärningspunkt, och M, N och P är punkter på sidorna AB, BC respektive CA som delar dessa sidor i samma förhållande (dvs $AM : MB = BN : NC = CP : PA = p : q$). Visa att O är medianernas skärningspunkt i triangeln som bildas utav linjerna AN, BP och CM.

20. Talföljder

4 augusti

1. Johan har glömt bort hur man räknar med rationella tal (men minns hur man adderar och multiplicerar heltal). Valentina har påminnt honom om att man kan skriva dem som par av heltal $(a, b) = \frac{a}{b}$ där b inte är noll. Definiera vilka räkneregler som gäller (hur adderar och multiplicerar man, och är några tal lika?). Varför är dina regler väldefinierade?

2. En talföljd är en följd av tal, t.ex. $1, 1, 2, 2, 3, 3, 4, 4, 5, 5, \ldots$. Talen som ingår kan tas från olika talmängder, t.ex. reella tal och heltal. Om inget annat anges så antar vi att våra talföljder består av reella tal. Ibland skriver man ut den (som ovan) om mönstret är uppenbart. Man kan också skriva en formel, t.ex. $a_n = 2^n$ istället för följden $2, 4, 8, 16, \ldots$. Vi studerar talföljden $1, -\frac{1}{2}, \frac{1}{3}, -\frac{1}{4}, \frac{1}{5}, -\frac{1}{6}, \ldots$.

(a) Skriv en formel för a_n.

(b) Det verkar som om följden närmar sig ett visst tal, vilket då?

(c) Kan vi hitta på en definition för vad det betyder att a_n närmar sig ett tal A då n blir stort (brukar skrivas $\lim_{n\to\infty} a_n$)?

(d) Använd definitionen för att visa att vår tidigare talföljd faktiskt går mot talet du angav.

3. Visa med din definition att talföljderna $a_n = n$ och $a_n = (-1)^n$ inte konvergerar (dvs inte går mot ett tal).

4. Om talföljden a_k konvergerar mot A och mot B, måste $A = B$ gälla?

5. * Om vi istället för att vara på $\mathbb{R}$ tar vår följd från ett metriskt rum (t.ex. $\mathbb{R}^2$) hur kan vi definiera att denna **punktföljd** går mot en punkt? Ett exempel kan vara $a_n = \left(\frac{1}{n}, \frac{1}{n^2}\right)$.

6. * Undersök punktföljden $a_n = \frac{1}{n}$ med diskreta respektive 3-adiska metriken på $\mathbb{Q}$. Konvergerar den (mot 0)? Hitta konvergenta punktföljder i båda metrikerna.

7. Låt a_n vara en ökande ($a_{k+1} > a_k$) följd av rationella tal så att $a_n < 100$ för alla n. Måste den konvergera mot ett rationellt tal? Bevis eller motexempel!

8. Låt a_n vara en ökande ($a_{k+1} > a_k$) följd av reella tal så att $a_n < 100$ för alla n. Måste den konvergera mot ett reellt tal? Bevis eller motexempel! Undersök det ett tag, om det inte lyckas, vänd på sidan.

För att visa egenskapen vi söker så måste vi förstå reella tal bättre. Speciellt så behöver vi en definition för reella tal!

Ett Dedekind-snitt D är en delmängd av de rationella talen som har följande egenskaper:

- $D \neq \mathbb{Q}$
- $D \neq \emptyset$
- Om $d \in D$ och $d' < d$ så vet vi att $d' \in D$.
- Om $d \in D$ så vet vi att det existerar något $d' \in D$ så att $d < d'$.

9. **(a)** Ge ett exempel på en mängd som inte är ett Dedekindsnitt, och ett som är ett exempel på ett Dedekindsnitt.

   (b) Visa att unionen och skärningen av två Dedekindsnitt är ett nytt Dedekindsnitt.

   (c) Är unionen/skärningen av oändligt många Dedekindsnitt ett nytt Dedekindsnitt?

10. **(a)** Vi vill använda Dedekindsnitt för att definiera reella tal från de rationella talen som vi redan känner till. Hur?

    (b) Hur kan du beskriva det Dedekindsnitt som motsvarar det reella talet $\sqrt{2}$?

    (c) Givet två reella tal a och b, hur kan vi definiera deras summa (via Dedekindsnitt)?

    (d) Givet två reella tal a och b, hur kan vi definiera deras produkt (via Dedekindsnitt)?

    (e) Givet två reella tal a och b, vad betyder det att $a < b$?

11. Låt a_n vara en ökande ($a_{k+1} > a_k$) följd av reella tal så att $a_n < 100$ för alla n. Måste den konvergera mot ett reellt tal? Bevis eller motexempel!

12. Låt a_n vara en talföljd så att $a_1 = 1$ och $a_{n+1} = \sqrt{1+a_n}$. Visa att den konvergerar och bestäm då vad den konvergerar mot.

13. Låt $a_0 = a$ och $b_0 = b$ vara positiva reella tal så att $a < b$. Vi definierar följderna

$$a_{n+1} = \sqrt{a_n b_n}, b_{n+1} = \frac{a_n + b_n}{2}.$$

Visa att a_n och b_n konvergerar mot samma tal.

21. Polynom II

4 augusti

Definition 1. Vi säger att polynomet $d(x)$ är *största gemensamma delare* till polynomen $p(x)$ och $q(x)$ om $d(x)$ delar både $p(x)$ och $q(x)$, samt om alla andra polynom $d'(x)$ som delar både $p(x)$ och $q(x)$ också delar $d(x)$.

Vi då använder beteckningen $\gcd(p(x), q(x)) = d(x)$ eller $\operatorname{sgd}(p(x), q(x)) = d(x)$ (*greatest common divisor* respektive *största gemensamma delare*).

För att beräkna sgd mellan två polynom kan man använda Euklides algoritm, precis som för heltal. Dvs. genomför upprepad polynomdivision där man i varje steg delar nämnaren i det föregående steget med resten från det föregående steget. Detta upprepas tills resten 0 erhålles, varvid sgd för de ursprungliga polynomen är den näst sista (dvs. sista icke-noll) resten.

1. Använd euklides algoritm för att bestämma:

   (a) $\operatorname{sgd}(x^4 - 17x^2 + 16, x^3 - 10x^2 + 17x + 28)$.

   (b) $\operatorname{sgd}(x^9 - 1, x^6 - 1)$.

2. Är sgd för två polynom unik? Bevisa eller ge motexempel!

3. Precis som för heltal finns det alltid polynom $a(x)$ och $b(x)$ så att $p(x)a(x) + q(x)b(x) = \operatorname{sgd}(p(x), q(x))$. Arbeta er baklänges genom euklides algoritm för att hitta $a(x)$ och $b(x)$ för $p(x) = x^4 - 17x^2 + 16$ och $q(x) = x^3 - 10x^2 + 17x + 28$.

   (Detta kallas för Bezouts identitet.)

4. Hitta ett polynom $p(x)$ så att $x^3 - x^2 - 1$ delar $p(x)$ och $x^2 - x$ delar $p(x) + 1$.

5. Ett reellt icke-konstant polynom $p(x)$ sägs vara *irreducibelt över* $\mathbb{R}$ om varje faktorisering $p(x) = q(x)r(x)$ i reella polynom $q(x)$ och $r(x)$ innebär att antingen $q(x)$ eller $r(x)$ är ett konstantpolynom.

(a) Ge ett exempel på ett polynom $p(x)$ av grad 2 som är irreducibelt över $\mathbb{R}$.

(b) Visa att alla reella icke-konstanta polynom $p(x)$ kan faktoriseras $p(x) = p_1(x)p_2(x)\cdots p_n(x)$ där alla polynomen p_i är irreducibla över $\mathbb{R}$.

(c) Visa att om $p(x)$ är irreducibelt över $\mathbb{R}$, $p(x)$ delar $q(x)r(x)$ och $p(x)$ inte delar $r(x)$ så delas $q(x)$ av $p(x)$. (Ledtråd: använd Bezouts identitet.)

(d) Visa att denna faktorisering är unik upp till ordningen på faktorerna (och muliplikation med en konstant).

6. Med heltal har vi både sgd och *minsta gemensamma multipel*, mgm. Definiera mgm för polynom.

7. Anta att polynomet $p(x)$ har nollställen i $a_1, a_2, \cdots, a_n$, där $a_i \neq a_j$ för $i \neq j$. Visa att i så fall delas $p(x)$ av $(x - a_1)(x - a_2)\cdots(x - a_n)$.

8. Visa att om $(x+1)$ delar $p(x^{100})$ så delar $(x-1)$ $p(x)$.

Extrauppgifter

9. **(a)** Polynomet $p(x)$ uppfyller $p(i) = i$ för $i = 1, 2, 3, 4$. Kan $p(5)$ vara något annat än 5?

(b) Hitta ett polynom $p(x)$ som har $p(1) = 2$, $p(2) = 4$, $p(3) = 8$, $p(4) = 16$.

10. Antag att $p(x) = a_n x^n + \cdots + a_1 x + a_0$ har udda grad, och att $a_i = a_{n-i}$ för alla $0 \leq i \leq n$. Visa att $p(-1) = 0$.

11. Visa att om $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$ och $r \neq 0$ är ett nollställe till $p(x)$, så har polynomet $q(x) = a_0 x^n + a_1 x^{n-1} + \cdots + a_n$ ett nollställe i $x = \frac{1}{r}$.

22. Inuti och utanför

5 augusti

1. Vi har tidigare studerat metriker. Låt oss för tillfället hålla oss till standardmetriken på $\mathbb{R}$ och $\mathbb{R}^2$. Studera bilden nedan med mängden M i $\mathbb{R}^2$ avbildad. I bilden nedan finns det olika typer av punkter, **inre punkter**, **yttre punkter** och **randpunkter** till M. Försök ge en definition på varje typ av punkt (gärna med hjälp av metriken).

2. Med din nya definition, bestäm randpunkterna och de inre punkterna till följande mängder M (alla utom Cantormängden är i $\mathbb{R}^2$).

(a) M är högra halvplanet, dvs punkter (x, y) så att $x \geq 0$.

(b) M består bara av origo.

(c) M består av allt utom origo.

(d) M är unionen av koordinataxlarna.

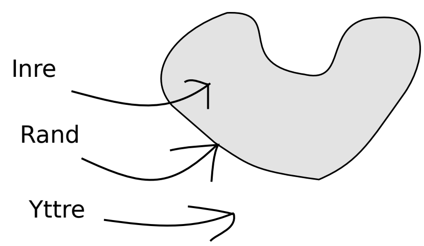
(e) M är de punkter $(x, 0)$ där $x = 1/n$ för något positivt heltal n.

(f) M är alla punkter (x, y) så att x och y båda ligger i $\mathbb{Q}$ (dvs är rationella tal).

(g) M är unionen av alla linjer genom origo med rationell lutning (dvs som kan skrivas på formen $y = kx$ där k är ett rationellt tal.

(h) Cantormängden från förra gången (där man startade med ett intervall och tog bort mittersta tredjedelen).

Vi säger att en mängd M som bara består av inre punkter är en **öppen** mängd och en som består av inre punkter och ALLA sina randpunkter är en **sluten** mängd.



3. Vi funderar nu på vad som händer om man tar unioner, snitt och komplement av öppna och slutna mängder. Se om det går att hitta en regel eller exempel på att olika saker kan hända (vi är alltså intresserade av om resultatet MÅSTE vara öppet eller slutet).

(a) Vad händer om vi tar en union av ändligt många öppna mängder?

(b) Vad händer om vi tar en union av ändligt många slutna mängder?

(c) Vad händer om vi tar komplementet av en öppen mängd?

(d) Vad händer om vi tar komplementet av en sluten mängd?

(e) Kan en mängd vara både öppen och sluten? Hur många sådana mängder finndet i $\mathbb{R}^2$?

4. Vad händer om vi istället tar oändligt många mängder i förra uppgiften?

5. Givet en mängd M så säger vi att mängden av alla inre punkter till M är mängden $IntM$ (från engelskans interior). På samma sätt så är mängden av alla inre punkter tillsammans med alla randpunkter till M mängden ClM (från closure, på svenska tillslutningen).

(a) Visa att $IntIntM = IntM$ och att $ClClM = ClM$.

(b) Hitta ett exempel på M i $\mathbb{R}$ där så många som möjligt av mängderna $M, IntM, ClM, IntClM, ClIntM, ClIntClM$ och $IntClIntM$ är olika.

6. Vi har nu tittat rätt mycket på öppna och slutna mängder. Ändrar sig vilka mängder som är öppna om vi byter från vanliga metriken till t.ex. Manhattan-metriken? Vad händer om vi tar godtycklig metrik? Kan vi hitta på en metrik på $\mathbb{R}^2$ så att alla mängder är öppna? Kan vi hitta på en så att ingen mängd är öppen?

7. Visa att varje öppen mängd på den reella linjen är en union av öppna intervall (dvs x sådana att $a < x < b$ för några värden på a och b).

8. I denna uppgift så är vi intresserade av att studera delmängder till $\mathbb{R}^2$ och $\mathbb{R}$ (som ju är rum med metriker, så vi kan definiera öppna och slutna mängder i dem) och se när det finns intressanta mängder som är både öppna och slutna samtidigt. Försök hitta vilken egenskap som är gemensam för dessa (och använd för att formulera en definition)!

(a) Reella tal x sådana att $x > 0$.

(b) Reella tal x sådana att $|x| > 1$.

(c) Rationella tal.

(d) Cirkeln i planet.

(e) Punkter i planet (x,y) så att $xy > 0$

(f) Punkter i planet (x,y) så att $xy \geq 0$

(g) De två reella talen 0 och 1.

9. Vi har tidigare kikat lite på talföljder, och definierat vad det betyder att a_n går mot A när n går mot ∞. Kan du göra den definitionen genom att använda öppna mängder istället för metriken?

10. En mängd T av delmängder till ett rum X kallas för en **topologi** om den har följande egenskaper:

1. $\emptyset \in T, X \in T$
2. För varje ändlig samling av element från T så ligger deras snitt i T.

3. För varje samling (dvs kan vara oändlig) av element i T så ligger deras union också i T.

Element i T brukar då kallas öppna mängder. Våra exempel på öppna mängder från metriker ger då alltid upphov till en topologi på ett rum.

11. Hitta ett rum X med en topologi T som INTE kan komma från en metrik (dvs vi kan aldrig hitta en metrik som har de mängderna som sina öppna mängder).

12. Hitta på en topologi på mängden $\{0,1\}$ så att varje följd konvergerar mot 0. Visa att följden med bara ettor även konvergerar mot 1. Du använder din definition på konvergens från uppgift 9.

13. Hitta på en topologi på $\mathbb{R}$ så att varje följd som inte innehåller 0 konvergerar mot ALLA tal utom 0 samtidigt.

14. Vilken egenskap behövs hos topologin för att man ska vara säker på att följder inte kan konvergera mot mer än en punkt? (Rum med denna egenskap kallas Haussdorf-rum, flera av de viktigaste topologiska rummen är INTE Haussdorf-rum).

23. Matematisk Yatzy

5 augusti

Algebra

1. (10) Lös ekvationssystemet

$$\begin{cases} 1 - x_1x_2x_3 = 0, \\ 1 + x_2x_3x_4 = 0, \\ 1 - x_3x_4x_5 = 0, \\ \dots \\ 1 + x_{48}x_{49}x_{50} = 0, \\ 1 - x_{49}x_{50}x_1 = 0, \\ 1 + x_{50}x_1x_2 = 0 \end{cases}$$

2. (20) För talen x, y, z gäller att $\frac{x}{y+z} + \frac{y}{z+x} + \frac{z}{x+y} = 1$. Vilka värden kan uttrycket

$\frac{x^2}{y+z} + \frac{y^2}{z+x} + \frac{z^2}{x+y}$ anta?

3. (30) Bestäm alla värden på a för vilka ekvationen $x^3 + x^2 = a$ kommer att ha tre reella rötter som bildar en aritmetisk talföljd.

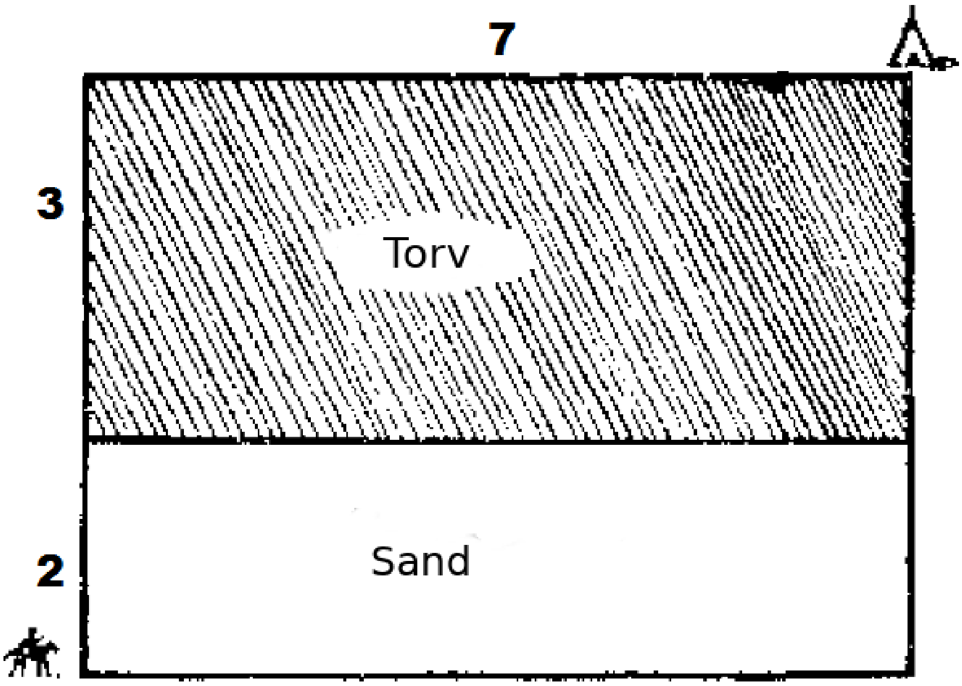
4. (40) Bestäm alla reella lösningar till ekvationssystemet:

$$\begin{cases} \dfrac{2x_1^2}{1+x_1^2} = x_2, \\ \dfrac{2x_2^2}{1+x_2^2} = x_3, \\ \dfrac{2x_3^2}{1+x_3^2} = x_1. \end{cases}$$

5. (50) Johan tänker på ett polynom $P(x)$ med icke-negativa heltalskoefficienter. Du får ställa frågor till Johan av typen "Vad är $P(k)$?" där k är ett heltal. Vilket är det minsta antalet frågor du behöver ställa för att garanterat veta Johans polynom? (Du får svara "oändligheten" på denna uppgift).

Geometri

1. (10) Det finns ett fält med bredden 5 och längden 7. Fältet är uppdelat i två delar: torv och sand med bredderna 3 respektive 2. Ryttaren kan förflytta sig genom torv dubbelt så fort som genom sand. Bestäm (rita ut) vägen från ett av fältets hörn till det motstående som tar kortast tid.



2. (20) Diagonalerna delar upp en parallelltrapets i fyra trianglar. S_1 och S_2 är areorna på de trianglarna som har en bas som sida. Uttryck parallelltrapetsens area S mha S_1 och S_2.

3. (30) I parallelltrapetsen $ABCD$ är basen $AD = 1$ och basen $BC = 2020$, medan $AB = 2019$. På linjen AD markerade man punkten E, som ligger på lika långt avstånd från C som från D. Bestäm DE.

4. (40) En liksidig triangel är inskriven i en kvadrat på så sätt att ett av triangelns hörn sammanfaller med ett av kvadratens hörn. Bestäm förhållandet mellan triangelns och kvadratens areor.

5. (50) Mittpunkten för cirkeln med radie 6 som tangerar sidorna AB, BC och CD på en likbent parallelltrapets $ABCD$ ligger på dess större bas AD. Basen BC är lika med 4. Bestäm avståndet mellan punkterna där cirkeln tangerar parallelltrapetsens icke-parallella sidor.

Talteori

1. (10) Skriv upp alla siffertripplar (a, b, c) sådana att $\overline{ab} \cdot c = a \cdot \overline{bc}$ och alla tre siffrorna är olika.

Med $\overline{ab}$ menas ett tvåsiffrigt tal vars siffror är a och b (i den ordningen).

2. (20) Skriv upp alla tal som är lika med summan av kuberna på deras siffror.

3. (30) Ett 29-siffrigt tal $X = \overline{a_1 a_2 a_3 \ldots a_{28} a_{29}}$ $(0 \leq a_k \leq 9, a_1 \neq 0)$ är givet. Man vet att för varje k så finns siffran a_k med i talet exakt a_{30-k} gånger (t.ex. om $a_{10} = 7$, så finns siffran a_{20} med 7 gånger). Bestäm siffersumman för talet X.

4. (40) Hur många positiva heltal x mindre än 10000 finns det för vilka $2^x - x^2$ är delbart med 7?

5. (50) David bakade en kladdkaka till alla sina kompisar. Han vet att det kommer bli totalt antingen p eller q personer vid bordet (p och q är relativt prima). I hur många bitar (som inte nödvändigtvis är lika stora) ska David dela upp kladdkakan i förväg för att garantera att alla gästerna ska kunna få lika mycket?

Kombinatorik

1. (10) Bland n riddare är varje par antingen vänner eller fiender. Varje riddare har exakt 3 fiender, dessutom är alltid fienderna till riddarens vänner också hans fiender. För vilka n är detta möjligt?

2. (20) För vilka n är alla koefficienterna i binomialutvecklingen utav $(a + b)^n$ udda?

3. (30) En uppdelning av ett positivt heltal A i summatermer $A = a_1 + a_2 + \cdots + a_{n-1} + a_n (n \geq 1)$ kallar vi för palindromialt om $a_1 = a_n, a_2 = a_{n-1}$ och generellt $a_i = a_{n+1-i}$ för $1 \leq i \leq n$. Bestäm antalet palindromiala uppdelningar för talet 2020.

4. (40) En halvmåne är en figur som man får utav två cirkelbågar. Vilket är det största antalet delar som 7 räta linjer kan dela upp halvmånens inre i?

5. (50) Det finns $2n+1$ batterier ($n>2$). Man vet att det finns en mer fungerande batterier än icke-fungerande, men man vet inte vilka som är vilka. Man sätter in två batterier i en ficklampa. Ficklampan lyser bara om båda batterierna är fungerande. Vilket är det minsta antalet försök som krävs för att ficklampan ska lysa?

Schackuppgifter

1. (10) En ”skadad kung” kan på ett drag bara ta sig till rutor som är ortogonalt angränsande. Den skakade kungen gjorde en rutt på ett 2020×2020-bräde. Den besökte varje ruta exakt en gång och kom tillbaka till startrutan. Låt oss bektrakta kungens rutt som periferin på en månghörning. Bestäm månghörningens area.

2. (20) På ett schackbräde finns det bönder i alla rutorna, totalt 32 svarta och 32 vita. En bonde kan bara ta en bonde av motsatt färg genom att röra sig diagonalt och ta den tagna bondens plats (de svarta bönderna kan bara ta nedåt-diagonalt, de vita bara uppåt-diagonalt). Bönderna kan inte röra sig på något annat sätt. Vilket är det minsta möjliga antalet bönder som kan vara kvar på schackbrädet?

3. (30) I en 8×8-tabell finns talen 1 och -1, ett tal i varje ruta. Betrakta alla sätt som man kan placera ett tetris-T över tabellen (figuren kan vridas och vändas, men måste läggas längst med rutornas gränser och inte sticka ut utanför brädet). Låt oss kalla placeringen *misslyckad* om summan av de fyra talen som figuren täcker inte är lika med 0. Bestäm det minsta möjliga antalet misslyckade placeringar.

4. (40) Låt oss kalla de 8 rutorna på en av schackbrädets diagonaler för staketet. Ett schacktorn har rört sig på brädet och besökt varje tom ruta en gång utan att gå på staketet (rutorna som tornet förflyttar sig över under ett drag räknas inte som besökta). Vilket är det största möjliga förflyttningar över staketet som tornet kan ha gjort?

5. (50) En kung besökte varje ruta på schackbrädet en gång och kom tillbaka till samma ruta som den startade på. (Kungen rör sig med vanliga regler, den kan på ett drag förflytta sig till en ortogonalt eller diagonalt angränsande ruta.) När man ritade ut kungens väg genom att förbinda mittpunkterna på rutorna i den ordningen den besökt dem med sträckor, så fick man en kurva som inte skär sig själv. Vilket är det största möjliga längden som kurvan kan ha? (En rutas sida är 1 längdenhet.)

24. Modulirum

7 augusti

Vi kommer idag att titta på så kallade modulirum, rum där varje punkt motsvarar ett matematiskt objekt. Ett exempel kan vara rummet av alla cirklar i planet. Där är varje punkt i det rummet en cirkel i planet. Ifall man funderar på det, så kan man ser att det rummet går att beskriva som $\mathbb{R}^2 \times \mathbb{R}_+$ ($\mathbb{R}_+$ står för positiva reella tal), för varje cirkel så måste vi dels bestämma dess centrum (en punkt i $\mathbb{R}^2$) och dels dess radie (ett positivt reellt tal). Rummet av cirklar i planet ser alltså ut som $\mathbb{R}^2 \times \mathbb{R}_+$ och är 3-dimensionellt. Vi har egentligen ingen definition på dimension än, men vi kan behandla det intuitivt (en cirkel är $1-$dimensionell osv) som antalet variabler som krävs för att beskriva en punkt i rummet.

1. Vilken dimension har följande rum?

(a) Trianglar i planet där ett hörn ligger på origo.

(b) Linjer genom origo i planet.

(c) Linjer genom origo i rummet.

(d) Linjesegment i planet (linjesegment får inte ha längden noll).

(e) Plan som går igenom origo i rummet.

(f) Ellipser i planet.

(g) Cirklar i rummet.

(h) Isometrier på linjen (en isometri är en funktion från ett rum till sig själv som bevarar avstånd, dvs $d(x,y) = d(f(x), f(y))$.

(i) Linjer i planet.

(j) Linjer i rummet.

(k) Plan i rummet.

2. Visa att rummet av linjer i $\mathbb{R}^3$ som går igenom origo och plan i $\mathbb{R}^3$ som går igenom origo inte bara har samma dimension, utan är lika (det finns ett naturligtsätt att gå från den ena till den andra).

3. Speciellt intressanta rum är rummet av linjer genom origo i planet (som kallas $\mathbb{R}P^1$, den projektiva linjen) och i rummet (som kallas $\mathbb{R}P^2$, det projektiva planet). Vi kommer att kika mer på dem (troligen kommande lektion) senare. Använd dessa rum (tillsammans med rummen vi känner till, $\mathbb{R}$, $\mathbb{R}^2$ etc. för att beskriva följande modulirum exakt.

(a) Cirklar i rummet.

(b) Rummet av alla ordnade par av linjer (dvs vi ser skillnad på linjerna) som korsar varandra med 90 graders vinkel i rummet.

(c) Rummet av alla linjer i rummet.

4. Visa att ifall vi tar bort en punkt från $\mathbb{R}P^2$ så får vi rummet av alla linjer i planet.

5. Låt oss kika på följande fysikaliska problemtyp. Vi har ett gäng metallstavar i planet (av samma längd). Änden på varje metallstav går att sätta fast i planet eller i änden på en annan metallstav (eller lämnas fri). Även om de är fastsatta så går de att rotera.

(a) Vad är rummet av en sådan metallstav som sitter fast i origo?

(b) Vad är rummet av två sådana metallstavar, där stav 1:s ena ände sitter i origo och stav 2 sitter i andra änden av stav 1?

6. Låt oss ta 4 stavar så att första staven sitter fast i den andra, andra i tredje osv. Låt oss sätta fast ändpunkterna i bordet i två givna punkter hyfsat långt ifrån varandra (lite över tre stavlängder från varandra=). Vad är dimensionen av rummet av sådana stavmaskiner? Vad är det rummet?

7. Låt oss ta två stavar till, ta sätta ihop dem, sätta fast ena ändpunkten i bordet och den andra i mitten av konstruktionen i förra uppgiften. Vad är dimensionen av detta rum och vad är det?

8. * Låt oss ta två stavar till, ta sätta ihop dem, sätta fast ena ändpunkten i bordet och den andra i mitten av konstruktionen i förra uppgiften. Vad är dimensionen av detta rum och vad är det? Denna uppgift kan upprepas flera gånger...

9. Visa att varje sådan metallstavskonstruktion kan beskrivas som en lösning till någon lämplig samling polynom (i flera variabler).

10. Låt $Gr_k(n)$ bestå av rummet av alla k-dimensionella plan genom origo i $\mathbb{R}^n$. Bestäm dimensionen av $Gr_k(n)$.

25. Affina avbildningar II

7 augusti

1. Visa att en affin transformation bevarar areaförhållanden mellan

(a) två trianglar där ett par sidor är parallella.

(b) två godtyckliga trianglar.

(c) två konvexa månghörnignar.

(d) två icke-konvexa månghörnignar.

2. I parallellogrammet $ABCD$ drar man en linje som är parallell med sidan AB och den skär sidan BC och diagonalen AC i punkterna N respektive K. Visa att trianglarna ADK och ABN har samma area.

3. Genom punkten O som ligger inuti triangeln ABC drog man tre linjer som är parallella med triangelns tre sidor. Triangeln delades då upp i tre trianglar och tre parallellogram. Areorna på trianglarna man fick vart 1, 2.25 och 4. Bestäm summan av areorna på parallellogrammen man fick.

4. **(a)** På sidorna AB, BC och AC i triangeln ABC satte man ut punkterna M, N, P samt symmetriska till dem kring respektive sidornas mittpunkter punkterna M', N' och P'. Visa att trianglarna MNP och $M'N'P'$ har samma area.

(b) På sidorna AB, BC och AC i triangeln ABC satte man ut punkterna M, N respektive P. Visa att om punkterna M', N' och P' på sidorna AC, AB respektive BC är sådana att $MM_1||BC, NN_1||CA, PP_1||AB$ så är areorna för trianglarna MNP och $M'N'P'$ lika stora.

5. På sidorna AB, BC, CD, DA i parallellogrammen $ABCD$ med arean S satte man ut punkterna C_1, D_1, A_1, B_1 så att $BC_1 = \frac{AB}{3}, CD_1 = \frac{BC}{3}, DA_1 = \frac{CD}{3}, AB_1 = \frac{DA}{3}$. Bestäm arean av fyrhörningen som bildas av linjerna AA_1, BB_1, CC_1, DD_1.

6. En linje som är parallell med diagonalen AC i fyrhörningen $ABCD$ som går igenom mittpunkten på diagonalen BD skär sidan AD i punkten E. Visa att sträckan CE delar fyrhörningen $ABCD$ i två delar som har lika stora areor.

7. Utav medianerna på en triangel byggde man ihop en ny triangel (varför är det möjligt att göra det?). Bestäm areaförhållandet mellan den nya triangeln och den ursprungliga.

Extrauppgifter

8. Visa att varje konvex fyrhörning förutom parallelltrapetser kan med hjälp av en affin avbildning avbildas på en fyrhörning där två motstående hörn är räta.

9. * Låt A vara en bijektiv avbildning på planet. Visa att A är affint om och endast om det är kontinuerligt (öppna mängders urbilder är öppna mängder) och avbildar linjer på linjer.

26. Fraktaler och hyperbolisk geometri

8 augusti

Dimension

Dimension är ett begrepp i matematiken som förekommer i olika delområden på olika sätt. Det är alltså definitivt inte självklart hur man definierar dimension (är t.ex. komplexa talen två eller endimensionella?). I denna del så ska vi kika litegrann på ett sådant dimensionsbegrepp för så kallade självsimilära mängder.

Vi säger att en begränsad (dvs den sticker inte ut mot oändligheten) mängd M i $\mathbb{R}^m$ är *självsimilär* om vi kan ta något reellt tal $0 < n < 1$, ta k stycken mindre kopior av M som är skalade med $1/n$ (dvs vi gör dem $1/n$ mindre på alla ledder) och sätta ihop dem så vi får tillbaka M igen. Exempelvis så är ett intervall $[0, 1]$ självsimilärt. Vi kan t.ex. ta två ($k = 2$) kopior av intervallet som vi gjort hälften så stora ($n = 2$) och sätta ihop för att få tillbaka ursprungliga intervallet.

1. Visa att en rektangel (inklusive innehåll) i planet är självsimilärt. Skriv upp några olika möjligheter för paret k och n.

2. Gör samma sak för en kub i rummet.

3. Kan du hitta något samband mellan k och n som ger dimensionen för objektet? Använd detta för att ge en definition för dimension (iallafall för självsimilära objekt).

4. **(a)** Visa att en hyperkub i $\mathbb{R}^4$ är självsimilär. Hyperkuben är de punkter (x, y, z, w) där $0 \leq x, y, z, w \leq 1$.

   (b) Prova din definition för att beräkna dimensionen för en hyperkub i R^4!

5. **(a)** Visa att *Cantormängden* är självsimilär (det var den där vi tog bort mittersta tredjedelen på ett intervall och fortsatte med småintervallen).

   (b) Vilken dimension har den (du kan använda dator/miniräknare/internet för att beräkna eventuella logaritmer odyl) enligt din definition?

(c) Vad händer om vi skulle ta bort mittersta femtedelen i varje steg istället?

6. En *Kochkurva* bildas på ett liknande sätt som Cantormängden. Man startar med ett intervall (låt oss ta intervallet $[0,3]$). Den mittersta tredjedelen tas bort och ersätts av två lika stora delar som bildar (delar av) en triangel som pekar uppåt. Vår figur består nu av 4 intervall av längd 1. Vi gör samma sak med dem (ta bort mittersta tredjedelen, ersätt med två sidor i en triangel som är lika stora som de dem började med och fortsätter.

(a) Rita några av stegen så du ser hur det fungerar (fortsätt rita tills det blir för svårt!).

(b) Hur lång är Kochkurvan? Hur stor area omsluter den?

(c) Visa att Kochkurvan är självsimilär.

(d) Vilken dimension har den (du kan använda dator/miniräknare/internet för att beräkna eventuella logaritmer odyl) enligt din definition?

7. En *Sierpinskitriangel* skapas genom att vi startar med en ifylld liksidig triangel, delar upp den i 4 lika stora liksidiga trianglar, och tar bort den mittersta. Vi upprepar processen med deltrianglarna och fortsätter på samma sätt.

(a) Vilken area har den?

(b) Vilken dimension har den?

8. Prova att skapa en egen självsimilär fraktal, det kanske går att ta någon av idéerna tidigare och använda i rummet på något sätt? Skriv ner konstruktionen och låt en kompis rita ut den och beräkna dimensionen! Ifall du vill ha lite mera utmaning, skapa den i det fyrdimensionella rummet!

Hyperbolisk geometri

Vi låter H vara den öppna disken med radie 1. Dvs (x,y) så att $x^2+y^2<1$. Hyperboliska punkter är punkter i denna disk. Vi säger att en hyperbolisk rät linje på denna disk är antingen en diameter eller en (del av) en cirkel som skär diskens kant med 90 graders vinkel (diameterfallet kan ses som en "cirkel" med centrum i oändligheten).

9. Skiljer sig geometrin på H (med de hyperboliska linjerna och punkterna) från den vanliga Euklidiska geometrin (med de vanliga linjerna och punkterna)?

10. Undersök vinkelsumman av en triangel i H (var sidor är segment av hyperboliska linjer). Är den konstant? I så fall, vilket värde antar den? Om inte, vad kan den variera emellan?

11. Hur kan man definiera parallella hyperboliska linjer i H?

12. Vi definierar avstånd på H så att avstånd inte ändras under reflektion i våra hyperboliska linjer. Reflektion i våra linjer ges av inversion i motsvarande cirkel. Vad skulle en rotation i H kring punkten p motsvara?

13. Låt p vara en punkt i H. Vi säger att en hyperbolisk cirkel med centrum i p som går igenom en punkt r, är alla punkter vi kan få genom att rotera r kring p. Bestäm de hyperboliska cirklarna i H.

14. En annan modell för H är övre halvplanet ((x,y) där $y > 0$). Då är linjer lodräta linjer tillsammans med cirklar med centrum på linjen $y = 0$. Förklara varför detta är samma geometri som H.

Det finns (minst) ett datorspel som utspelar sig på hyperboliska planet, "hyperrogue". Det finns gratis på
https://roguetemple.com/z/hyper/
Det kan ge någon form av intuition för att det hyperboliska planet är "större" än det vanliga.

27. Polynom III (Normal)

8 augusti

Algebrans fundamentalsats. Varje komplext icke-konstant polynom $p(x)$ har minst ett komplext nollställe.

1. Visa att alla polynom $p(x)$ kan skrivas på formen $c(x - x_1)\cdots(x - x_n)$ där x_i är de (möjligtvis komplexa) nollställena till $p(x)$ och $\deg p(x) = n$.

2. Visa att denna *faktorisering* av $p(x)$ är unik (upp till ordningen på faktorerna).

3. Visa att om polynomen $p(x)$ och $q(x)$ båda har grad som mest n och det finns $n+1$ distinkta tal $x_0, x_1, x_2, \cdots, x_n$ för vilka $p(x_i) = q(x_i)$ så är $p(x) = q(x)$.

4. Antag att $p(n) = n^2$ för alla $n = 1, 2, 3, 4, \cdots$. Måste $p(x) = x^2$?

5. Antag att polynomet $p(x)$ har reella koefficienter och grad 6 samt har 6 distinkta nollställen. Bevisa att det omöjligt kan vara så att precis ett av nollställena är icke-reellt.

6. Antag att polynomet $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$ av grad n har nollställen $x_1, x_2, \cdots, x_n$. Visa att:

(a) $(-1)^n \frac{a_0}{a_n} = x_1 x_2 \cdots x_n$.

(b) $-\frac{a_{n-1}}{a_n} = x_1 + x_2 + \cdots + x_n$.

(c) generellt så är summan av alla produkter av $0 < r \leq n$ nollställen lika med $(-1)^r \frac{a_{n-r}}{a_n}$.

Dessa formler kallas *Vietas formler*.

7. Man vet att $x = 10$ är en rot till ekvationen $x^2 + px + 2 - 5\sqrt{3} = 0$. Bestäm den andra roten.

8. Erland skrev upp fem heltal på tavlan – koefficienterna i ett andragradspolynom och dess nollställen. Johan suddade bort ett av talen. Då stod talen 2, 3, 4 och -5 kvar på tavlan. Vilket tal suddade Johan bort?

9. Antag att a, b, c är reella tal och att $a + b + c = 0$. Visa att $a^3 + b^3 + c^3 = 3abc$.

10. Antag att i, j, k är alla lösningarna till ekvationen $5x^3 - 11x^2 + 7x + 3 = 0$.

(a) Vad blir $i^2 + j^2 + k^2$?

(b) Vad blir $i^3 + j^3 + k^3$?

Extrauppgifter

11. Visa att om $p(x)$ är ett polynom med grad 2 så finns det polynom $q(x)$ och $r(x)$, båda med grad 2, så att $p(x)p(x+1) = q(r(x))$.

12. Hitta alla polynom $p(x)$ för vilka likheten $p(x^2) + p(x)p(x+1) = 0$ håller för alla x.

13. Polynomet $p(x)$ har grad n och egenskapen att $p(k) = \frac{k}{k+1}$ för alla $k = 0, 1, 2, \cdots, n$. Vad är $p(n+1)$?

28. Polynom III (Svårare)

8 augusti

Sats. Varje polynom med komplexa koefficienter med grad minst ett har åtminstone en komplex rot.

Betrakta ett polynom av grad n med komplexa koefficienter (som också kan vara reella tal), som vi skriver som $P(z) = z^n + a_{n-1}z^{n-1} + \cdots + a_1 z + a_0$, eftersom det alltid går att förkorta med den största koefficienten utan att förändra rötterna. Låt oss betrakta polynomet som en funktion från $\mathbb{C}$ till $\mathbb{C}$. Låt oss flytta runt en punkt z i det komplexa planet och betrakta hur $P(z)$ flyttar sig på samma komplexa plan under tiden.

Om $a_0 = 0$ så har polynomet roten $z = 0$, så vi kommer bara att studera fallen då $a_0 \neq 0$. Vi kallar punkten z^n på det komplexa planet för *Damen* och punkten $P(z)$ för *Hunden*. Oavsett var punkten z är så är avståndet mellan Damen och Hunden lika med $|P(z) - z^n|$.

1. **(a)** Talet z börjar sin väg i det komplexa planet i en reell punkt R, flyttar sig längs med en cirkel med radien R motsols och kommer i slutändan tillbaka till samma punkt. Hur kommer Damen att flytta sig under tiden?

(b) Vi kallar talet L för *kopplets längd* om det är det största möjliga avståndet mellan Damen och Hunden för ett givet värde på R. Visa att för $R > 1$ så är kopplets längd begränsat av

$$L \leq (1 + |a_{n-1}| + |a_{n-2}| + \cdots + |a_1| + |a_0|)R^{n-1}$$

(c) Vilken radie ska man ta på cirkeln som z rör sig på för att kopplets längd ska vara mindre än R^n som är Damens avstånd från punkten 0?

Så om z rör sig runt origo på en cirkel med en radie som bestämdes i uppgift 1(c) så kommer Hunden liksom Damen att vandra n gånger runt origo. Låt oss minska den ursprungliga cirkelns radie. Radien för Damens cirkel kommer också att minska men hon kommer fortfarande att vandra runt origo n gånger.

Hundens bana kommer att ändras kontinuerligt, det vill säga för en "liten" förändring på R så kommer banan att förändras obetydligt. Dock om R är nära 0, så kommer banan att vara nära punkten a_0.

2. Visa att för $R < 1$ så kommer Hunden befinna sig på ett avstånd som är som mest $L \leq (1 + |a_{n-1}| + |a_{n-2}| + \cdots + |a_1| + |a_0|)R$ ifrån punkten a_0.

Då finns det ett litet värde på radien för vilken Hunden inte kommer vandra runt origo en enda gång. Så när R minskar, så minskar antalet gånger som Hunden går runt origo till noll.

3. Argumentera för att för något värde på radien så kommer Hunden att bli tvungen att vandra genom origo.

Satsen är bevisad!

4. Visa att varje polynom med komplexa koefficienter av grad $n \geq 1$ faktoriseras i n linjära faktorer med komplexa koefficienter på ett unikt sätt upp till ordning och multiplikation med konstanter.

5. Visa att varje polynom med reella koefficienter faktoriseras i (högst) andragradstermer med reella koefficienter på ett unikt sätt upp till ordning och multiplikation med konstanter.

6. Visa att varje polynom av udda grad som har reella koefficienter alltid har en reell rot.

Extrauppgifter

7. Antag att $p(n) = n^2$ för alla $n \in \mathbb{N}_+$. Måste $p(x) = x^2$?

8. Hitta det polynom av högst grad som delar $x^7 - 1$ men ej delar $x^6 - 1$ och har 1 som sin första koefficient.

9. m och n är positiva heltal $a^m = 1$ och $a^n = 1$ och $\text{sgd}(m,n) = 1$. Visa att $a = 1$.

10. Det finns två polynom $P(x)$ och $Q(x)$ var grad är 10 och högsta koefficienten är lika med 1. Man vet att ekvationen $P(x) - Q(x)$ inte har några reella rötter. Visa att ekvationen $P(x+1) = Q(x-1)$ har åtminstone en reell rot.

11. Det finns två polynom $P(x)$ och $Q(x)$, båda av positiv grad, så att $P(P(x)) = Q(Q(x))$ och $P(P(P(x))) = Q(Q(Q(x)))$. Stämmer då nödvändigtvis likheten $P(x) = Q(x)$?

29. Algebraiska kurvor och ytor

9 augusti

1. Vi säger att en algebraisk mängd i planet är en mängd som är gemensamt nollställe till ett ändlligt antal polynom.

   (a) Visa att varje mängd i planet som består av två punkter är algebraisk.

   (b) Visa att unionen av två algebraiska mängder är algebraisk.

2. Rita upp nollställemängden till polynomet $p(x,y) = xy$. Fundera på vad som händer när vi istället för att lösa $p = 0$ försöker lösa $p = \varepsilon$ där ε är ett litet positivt tal. Försök att fundera först innan du börjar räkna.

3. Gör samma sak för polynomet $p(x,y) = y(x^2 + y^2 - 1)$.

Om vi har en kurva i planet givet av ett polynom av grad d som inte har några singulariteter (dubbelpunkter, självkorsningar etc) så säger Harnacks olikhet att det högst kan finnas $\frac{(d-1)(d-2)}{2} + 1$ komponenter. Talet d är polynomets grad.

4. Kan du konstruera en ickesingulär fjärdegradskurva i planet med 4 komponenter?

Bezouts sats Denna sats säger att om du har en kurva av grad d_1 och en kurva av grad d_2 så innehåller skärningen antingen oändligt många punkter eller högst

$d_1 d_2$ stycken (egentligen exakt $d_1 d_2$ om man räknar multiplicitet och komplexa skärningar).

5. Visa Bezouts sats i fallet $d_1 = 1$.

6. Hur kan komponenterna från din fjärdegradskurva möjligen vara placerade? Går det att ha 4 "cirklar" inuti varandra? Försök hitta (med bevis!) alla sätt som en fjärdegradskurva kan se ut på.

Gratisprogrammet SURFER finns på
imaginary.org/program/surfer

Det går att använda för att rita upp algebraiska ytor i rummet där man kontinuerligt kan variera vissa parametrar.

Följande uppgifter är inte givna i någon speciell ordning, välj en som låter intressant och experimentera, gärna med SURFER som hjälpmedel. Fundera först vad du tror gäller och använd programmet sen.

7. Vad händer med $t = xyz$ när vi låter t variera från 0 till 1. Vad startar vi med och hur kommer det se ut efteråt.

8. Vi vill låta polynomet p ha unionen av en cylinder och ett plan som nollställe. Vad händer när vi sätter $p = t$ och låter t variera?

9. Försök konstruera kurvor med maximalt antal komponenter (enligt Harnack). Hur långt upp i graderna kommer du?

10. Hur kan en ickesingulär femtegradskurva se ut?

11. Vi vill låta polynomet p ha unionen av två cylindrar som nollställe. Vad händer när vi sätter $p = t$ och låter t variera?

12. Prova att ta några ytor i rummet, titta på deras union som nollställemängden till ett polynom p med surfer och kolla vad som händer när vi låter t gå från 0 till 1 i ekvationen $p = t$.

13. Hilberts sextonde problem ställer frågan hur en ickesingulär sjättegradskurva i planet med maximalt antal komponenter kan se ut. Finns det någon/några situationer vi vet inte kan hända?

14. Samma problem har lösts också för grad 7. Grad 8 är fortfarande ett öppet problem (så ingen vet lösningen).

15. SURFER har ett bibliotek över intressanta ytor, både de som mest ser roliga ut (ett hjärta etc) och de som är intressanta vetenskapligt (maximalt antal singulariteter t.ex.). Undersök biblioteket och prova gärna att ändra på polynomen och se vad som händer!

30. Polynom IV

9 augusti

1. Antag att $p(x) = c_n x^n + \cdots + c_1 x + c_0$ är ett polynom med heltalskoefficienter, $\frac{a}{b}$ är ett bråk förkortat så långt som möjligt och att $p(\frac{a}{b}) = 0$. Visa att a i så fall delar c_0 och b delar c_n.

2. Visa att ekvationen $x^4 - x^3 + 7x^2 + 6x + 8 = 0$ inte har några rationella lösningar.

Definition 1. Ett polynom $p(x)$ med heltalskoefficienter sägs vara *irreducibelt över* $\mathbb{Z}$ om det inte går att skriva $p(x) = q(x)r(x)$ där $q(x)$ och $r(x)$ är icke-konstanta polynom med heltalskoefficienter.

3. Antag att $p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_0$ är ett polynom med heltalskoefficienter. Bevisa att om det finns ett primtal p så att

- p delar a_i för alla $i = 0, 1, \cdots, n-1$
- p delar inte a_n
- p^2 delar inte a_0

så är $p(x)$ irreducibelt över $\mathbb{Z}$.

Ledning: Antag att $p(x) = q(x)r(x)$där $q(x) = b_s x^s + b_{s-1} x^{s-1} + \cdots + b_0$ och $r(x) = c_t x^t + c_{t-1} x^{t-1} + \cdots + c_0$ är heltalspolynom.

(a) Bevisa att det alltid finns ett tal $0 \leq i \leq s$ $(0 \leq j \leq t)$ så att i är det minsta tal så att p inte delar b_i (c_j).

(b) Vilka värden kan i och j ha? Studera a_{i+j}:s delbarhet med p och försök få fram en motsägelse.

Denna sats kallas *Eisensteins kriterium*.

4. Visa att polynomet $x^{101} + 99x^{100} + 102$ är irreducibelt över $\mathbb{Z}$.

5. Visa att för alla positiva heltal n finns ett polynom av grad n som är irreducibelt över $\mathbb{Z}$.

6. Bevisa att om $p(x)$ är ett polynom med heltalskoefficienter och a, b är två olika heltal så delar $a - b$ talet $p(a) - p(b)$. (Ledtråd: visa att $a^n - b^n = (a-b)(a^{n-1} + a^{n-2}b + \cdots + ab^{n-2} + b^{n-1})$.)

7. Visa att om $P(2)$ är delbart med 5 och $P(5)$ är delbart med 2 så är $P(7)$ är delbart med 10, där $P(x)$ är ett polynom med heltalskoefficienter.

8. **(a)** Bevisa att $\text{sgd}(x^n - 1, x^m - 1) = x^{\text{sgd}(m,n)} - 1$ för två positiva heltal n och m.

(b) Hitta det polynom $q_n(x)$ som delar $x^n - 1$, där $n > 1$ är ett heltal, men inte har någon gemensam faktor $x^{n-1} - 1$.

(c) Bevisa att om $p(x)$ är irreducibelt över $\mathbb{Z}$ så är också $q(x) = p(x+1)$ irreducibelt över $\mathbb{Z}$.

(d) Bevisa att om p är ett primtal så är $q_p(x)$ irreducibelt över $\mathbb{Z}$.

Extraupgifter

9. Visa att det inte finns något polynom $p(x)$ med heltalskoefficienter så att $p(2019) = 2019$ och $p(2016) = 2017$.

10. Antag att polynomet $p(x) = ax^3 + bx^2 + cx + d$, där a, b, c, d är heltal, bara har reella nollställen samt att ad är udda och bc är jämnt. Visa att i så fall har $p(x)$ minst en icke-rationell rot. (Ett icke-rationellt tal är ett tal som inte kan skrivas som en kvot $\frac{x}{y}$ av två heltal x och y.)

11. Bestäm sgd utav $(x-1)^{103} + 1$ och $(x+1)^{101} - 1$.

12. Visa att polynomet $x^{101} + 101x^{100} + 102$ är irreducibelt över $\mathbb{Z}$.

13. $P(x)$ och $Q(x)$ är två polynom med samma uppsättning av heltalskoefficienter, som eventuellt kommer i olika ordning. Visa att $P(2019) - Q(2019)$ är delbart med 1009.

14. $P(x)$ är ett polynom med heltalskoefficienter. Ekvationen $P(x) = 2$ har tre heltalslösningar. Visa att ekvationen $P(x) = 3$ inte har några heltalslösningar.

15. Låt n vara ett positivt heltal. På $2n+1$ kort står var sitt nollskilt heltal, summan av alla talen är också nollskilt. Man vill ersätta stjärnorna i uttrycket

$$*x^{2n} + *x^{2n-1} + \cdots + *x + *$$

så att det polynomet man erhåller inte har några heltalsrötter. Går detta alltid att göra?

31. Mattedrabbning lila

10 augusti

Lag 1 - Lag 2

1. I en schackturnering spelar 30 lag mot varandra, så att alla spelar mot alla. För vinst ges 1 poäng, oavgjort ger $\frac{1}{2}$ poäng och förlust ger 0 poäng. Vilket är det största möjliga antalet schackspelare som efter turneringens slut kunde ha haft exakt 5 poäng?

2. Visa olikheten $a^2 + b^2 + c^2 + 2abc < 2$ om a, b, c är sidorna i en triangel med omkretsen 2.

3. Man har skrivit upp talen från 1 till 1000. Hur många av dessa tal kan skrivas som en differens mellan två positiva kvadrattal?

4. I triangeln ABC är vinkeln B lika med $60°$. Bisektriserna AK och CE skär varandra i punkten O. Visa att $OK = OE$.

5. Hitta alla polynom $p(x)$ som uppfyller $p(x) = \frac{p(x+1)-p(x-1)}{2}$ för alla x och $p(0) = 0$.

6. Aron och Isak spelar följande spel på en graf. Aron ställer sig i en av noderna, efter det ställer Isak sig i någon av de andra noderna. Därefter fördelar de grafens alla noder enligt följande regel: den som får noden är den som behöver gå på först kanter för att komma fram till den. Om det är lika många kanter, så får ingen noden. Finns det någon graf så att Isak garanterat kan få mer än hälften av alla noder i slutet av spelet?

7. I triangeln ABC på fortsättningen av sidan CB bortom B har man markerat punkten D så att $AB = BD$. Låt M vara mittpunkten på sidan AC. Bisektrisen till vinkeln ABC skär linjen DM i punkten P. Visa att vinkeln BAP är lika med vinkeln ACB.

8. Givet $k \in \mathbb{N}^+$ visa att det finns ett $a \in \mathbb{N}$ så att talföljden $a, a+k, a+2k, a+3k, \ldots$ innehåller oändligt många primtal.

Lag 3 - Lag 4

1. Parablerna $y = x^2 + bx + c$ och $x = y^2 + by + c$ har exakt en punkt där de sammanfaller. Visa att om b är ett udda tal, så är c ett kvadrattal.

2. Går det att dela upp alla positiva heltal i 3 icke-tomma mängder så att för alla par av tal x och y från olika mängder så tillhör talet $xy + x + y$ den tredje mängden?

3. En triangel ABC är given. De vidskrivna cirklarna tangerar sidorna AB och AC i punkterna P och Q. Sträckorna CP och BQ skär varandra i punkten K, medan mittpunktsnormalerna till dessa sträckor skär varandra i punkten L. Visa att om punkterna A, K och L ligger på en linje så är triangeln ABC likbent.

4. Visa att produkten av tre på varandra följande positiva heltal inte kan vara ett kvadrattal.

5. I $n - 1$ rutor utav ett $n \times n$-bräde sitter var sin skalbagge. En skalbagge vaktar alla ortogonalt angränsande rutor. Visa att de finns en skalbagge som kan förflytta sig till en (ortogonal) grannruta som är ledig och som inte vaktas av en annan skalbagge.

6. För positiva heltal $a < b \leq c < d$ vet man att $ad = bc$ och $\sqrt{d} - \sqrt{a} \leq 1$. Visa att a är ett kvadrattal.

7. Jonathan tänker på n olika punkter som ligger inuti en given halvdisk. Elias kan peka på en punkt i planet och då kommer Jonathan berätta avståndet från den till den närmsta av de punkterna som han tänker på. Hur kan Elias på $2n + 1$ frågor bestämma minst en av punkterna som Jonathan tänker på (att bestämma innebär att få ett svar på frågan som är exakt 0)?

8. På ett 100×100-bräde står ett vitt torn och ett svart torn som inte hotar varandra. Tornen turas om att göra drag, det vita börjar. Ett torn kan inte flytta sig till en ruta där det redan har varit eller så att det blir hotat av det andra tornet. Den som inte kan göra ett drag förlorar. Vem vinner om båda spelar optimalt?

Inversion specialkurs

32. Cirklar är linjer

2 augusti

Följande 4 uppgifter följer ett intressant mönster.

1. Låt $\omega_{1,2,3,4}$ vara fyra cirklar så att ω_1 och ω_3 tangerar ω_2 och ω_4. Visa att de fyra tangeringspunkterna ligger på en cirkel.

2. Låt $\omega_{1,2,3}$ vara tre cirklar och l_4 en linje så att ω_1 och ω_3 tangerar ω_2 och l_4. Visa att de fyra tangeringspunkterna ligger på en cirkel.

3. Låt ω_1 och ω_3 vara två cirklar och l_2 och l_4 två linjer så att båda cirklarna tangerar de båda linjerna. Visa att de fyra tangeringspunkterna ligger på en cirkel.

4. Låt ω_1 och ω_2 vara två tangenta cirklar och låt l_3 och l_4 vara två parallella linjer så att ω_1 och l_4 såväl som ω_2 och l_3 är tangenta. Visa att de tre tangeringspunkterna ligger på en linje.

Dessa uppgifter är så pass lika varandra att vi skulle vilja kalla alla fyra för exakt samma uppgift, den enda som verkar avvika från de andra är uppgift 4. Vi kan lite mer tydligt visa denna likhet genom att transformera en av uppgifterna till en annan. Betrakta alltså uppgift 1 och ”blås upp” cirkeln ω_4 så att radien går mot oändligheten men så att den alltid tangerar ω_1 och ω_3. I denna process kommer cirkeln ω_4 att gå mot linjen l_4 i uppgift 2, och de fyra tangeringspunkterna kommer fortfarande att ligga på en cirkel!

Tanken är alltså att betrakta en linje som en cirkel med oändlig radie, denna ide kan vid flera tillfällen vara användbar. Mer formellt introducerar vi ”punkten vid oändligheten” betecknat P_∞. Vi säger att denna punkt ligger på alla linjer men inte på någon cirkel. Planet tillsammans med P_∞ utgör ”det utökade” planet, som har den trevliga egenskapen att tre godtyckliga punkter definierar antingen en cirkel eller en linje. I det utökade planet skiljer vi inte heller på cirklar och linjer, utan betraktar de som samma typ av objekt, ”clinjer” (”clines” på engelska). Vi säger att två clinjer tangerar varandra om de endast har en gemensam punkt, så två parallella linjer tangerar alltså varandra i P_∞.

Vi kan nu översätta alla fyra uppgifter till samma uppgift i det utökade planet på följande sätt.

5. Låt $\Gamma_{1,2,3,4}$ vara fyra clinjer i det utökade planet så att Γ_1 och Γ_3 tangerar Γ_2 och Γ_4. Visa att de fyra tangeringspunkterna ligger på en clinje.

Den intressantaste versionen är uppgift 4. Märk att de två parallella linjerna har en tangeringspunkt i P_∞. För att visa att de fyra tangeringspunkterna ligger på en clinje ska vi alltså visa att de tre andra tangeringspunkterna ligger på en linje!

33. Inversioner runt en punkt

3 augusti

När vi nu är bekväma med det utökade planet kan vi definiera inversion med avseende på en cirkel.

Definition 1. Om ω är en cirkel med mittpunkt O och radie r är inversen av punkten P med avseende på cirkeln ω den punkten P^* på strålen OP som uppfyller $OP \cdot OP^* = r^2$.

Från definitionen framkommer det även att inversen av O är P_∞ och vice versa. Med detta i åtanke blir inversion definierat på alla punkter av det utökade planet. Det är även uppenbart att inversen av inversen av en punkt är punkten själv. Inversion är alltså en involution och således en bijektion!

När vi inverterar runt en cirkel spelar radien sällan någon roll. Att ändra radien av cirkeln vi inverterar med avseende på resulterar endast en i förstoring eller förminskning av bilden vi får. På grund av detta pratar vi oftast om inversion runt en punkt, och menar inversion med avseende på någon cirkel med denna punkt som mittpunkt.

1. Invertera runt punkten O. Visa att $\angle OAB = \angle OB^*A^*$.

Den föregående satsen är mycket hjälpsam och kommer bland annat vara användbar för att visa följande satser.

2. Låt l vara en linje genom O. Visa att inversion runt O tar linjen till sig själv.

3. Låt l vara en linje ej genom O. Visa att inversion runt O tar linjen till en cirkel genom O. (Tips: Tänk på punkten på l som ligger närmast O.)

4. Låt ω vara en cirkel genom O. Visa att inversion runt O tar cirkeln till en linje ej genom O.

5. Låt ω vara en cirkel ej genom O. Visa att inversion runt O tar cirkeln till en annan cirkel ej genom O. (Tips: Betrakta den diametern som går igenom O.)

Dessa uppgifter visar den viktigaste egenskapen av inversion: inversion tar clinjer till clinjer! Tack vare att inversion är en bijektion tas även tangenta clinjer till tangenta clinjer! Sammanfattningsvis gör inversion runt punkten O följande:

- Linjer genom $O \leftrightarrow$ Linjer genom O.
- Linjer ej genom $O \leftrightarrow$ Cirklar genom O.
- Cirklar ej genom $O \leftrightarrow$ Cirklar ej genom O.

Ett varningsord: Om cirkeln ω_1 tas till cirkeln ω_2 tas mittpunkten av ω_1 inte till mittpunkten av ω_2 Detta är ett mycket typiskt misstag som många faller på.

34. Inversioner runt en punkt

4 augusti

De enda punkterna som bevaras under inversion med avseende på en cirkel är punkterna på själva cirkeln. Vi vet även att linjer genom O tas till sig själva under inversion runt O. Vilka andra clinjer tas till själva under inversion? Den frågan får vi snart svar på.

Definition 1. Låt ω_1 och ω_2 vara två cirklar med mittpunkter O_1 och O_2 samt skärningspunkter A och B. Om $\angle O_1AO_2 = \angle O_1BO_2$ är de två cirklarna ortogonala.

1. Låt ω_1 och ω_2 vara två ortogonala cirklar. Visa att ω_2 tas till själv under inversion med avseende på ω_1.

Ibland vill man jobba med längder, då kan följande uppgift vara användbar.

2. Invertera runt en cirkel med mittpunkt O och radie r. Visa följande formel:

$$A^*B^* = r^2 \cdot \frac{AB}{OA \cdot OB}.$$

Triangelcentrum visar sig ha bekväma egenskaper under inversion.

3. Låt triangeln ABC ha omcentrum O, orthocentrum H och incentrum I. Visa följande under inversion runt A.

- O^* är spegelbilden av A på linjen B^*C^*.
- I^* är excentrum av triangeln AB^*C^*.
- A är excentrum av triangeln $H^*B^*C^*$.

35. Uppgifter med inversion

7 augusti

Inversion är ett tveeggat svärd, inversion runt en illa vald punkt kan oftast försvåra uppgifter istället för att förenkla dem! Det är därför viktigt att veta när det är lämpligt att invertera.

- Inversion kan ta cirklar till linjer, dessa är oftast enklare att jobba med. Om många linjer går igenom samma punkt, invertera runt den punkten!
- Inversion tar tangenta clinjer till tangenta clinjer. Uppgifter med många tangenta linjer och cirklar kan oftast ritas om till en finare konfiguration med hjälp av inversion.
- Inversion byter plats på vinklar, skicklig användning av detta kan skapa trevligare konfigurationer.

1. Triangeln ABC har en rät vinkel vid A. Cirklarna ω_1 och ω_3 genom A har mittpunkter på sidan AB. Cirklarna ω_2 och ω_4 genom A har mittpunkter på sidan AC. Visa att de 4 punkterna där två av cirklarna skär varandra ligger på en cirkel.

2. Låt Cirklarna ω_1 och ω_3 tangerar varandra vid P. Cirklarna ω_2 och ω_4 tangerar också varandra vid samma punkt P. Låt de 4 punkterna där två av cirklarna skär varandra vara A, B, C och D. Visa att:

$$\frac{AB \cdot BC}{AD \cdot DC} = \frac{PB^2}{PD^2}.$$

3. Låt $ABCD$ vara en konvex fyrhörning där diagonalerna skär varandra i en rät vinkel vid punkten P. Visa att de 4 spegelbilderna av P på fyrhörningens sidor ligger på en cirkel.

4. Låt ω vara en halvcirkel med diameter AB. Linjen l skär ω i punkterna C och D samt linjen AB i punkten M så att $MB < MA$ och $MD < MC$. Låt K vara skärningspunkten av cirklarna runt AOC och BOD. Visa att $\angle OKM = 90°$.

5. Linjen l skär cirkeln ω i A och B. Låt cirkeln Γ tangera AB vid T och tangera ω internt vid S. Visa att linjen TS delar randen AB i två lika delar.

6. Cirkeln ω har diameter PQ. Cirkeln Γ tangerar ω internt och tangerar PQ vid punkten C. Låt AB vara den tangenten av Γ som är rätvinklig mot PQ och så att A ligger på ω och B ligger på PQ. Visa att AC är bisektrisen av $\angle BAP$.

7. A, B och C ligger på en linje i den ordningen. Cirkeln ω_1 har diameter AB och cirkeln ω_2 har diameter BC. Cirkeln ω_3 tangerar ω_1, tangerar linjen genom C som är rätvinklig mot AC och tangerar ω_2 vid punkten T. Visa att linjen AT tangerar ω_2.

8. Fyrhörningen $ABCD$ har en omskriven cirkel med mittpunkt i O och en inskriven cirkel med mittpunkt i I. Ytterbisektriserna av $\angle A$ och $\angle C$ skär ytterbisektriserna av $\angle B$ och $\angle D$ i punkterna Q, R, S och T. Visa att $QRST$ ligger på en cirkel med mittpunkt på linjen OI.

IV Lektionsmaterial programmeringsgruppen Babbage

36. Legorobotar

2 augusti

Idag ska vi installera utvecklingsmiljön för att programmera Legorobotar med Python. Ni kanske har programmerat legorobotar tidigare med grafiska block, men man kan faktiskt programmera dem med Python och andra textuella programmeringsspråk också. Vi kommer använda oss av programmet Visual Studio Code, med ett tillägg för EV3 Mindstorms.

Vi kommer gå igenom allting från grunden så ni behöver inga förkunskaper.

36.1 Anropa funktioner utan returvärde

För att anropa en funktion skriver man dess namn och sen parenteser som innehåller dess argument. Precis som att man i matematiken kan skriva f(x) eller h(x). En stor skillnad är att i matten döps de flesta funktioner till en enda bokstav, medan de i programmeringen brukar ha längre namn som `print()` eller `reset_angle()`.

≫ Första uppgiften är att anropa funktionen print() som skriver ut saker på skärmen. Dess argument är en textsträng som ska skrivas ut på skärmen (på datorn, inte roboten). Textsträngar lägger man alltid mellan citatstecken i python, såhär: `"Hejsan!"` Du kan även skriva ut tal eller andra datatyper med printfunktionen. Den är praktisk när man felsöker sina program.

Många funktioner brukar grupperas i något som heter klasser. Till exempel alla funktioner som har något med ev3-klossen att göra ligger i klassen brick. Sen finns det underklasser till den, till exempel allting som har med högtalaren i ev3-klossen att göra ligger i en klass som heter sound. Klasser och funktioner separerar man med punkt i python. Så för att spela upp ett ljud kan man skriva `brick.sound.beep()`.

Det går att skriva ut saker på ev3-enhetens skärm också, då använder man funktionen text() i klassen display som ligger i brick-klassen. För att skriva ut något skriver man `brick.display.text("En fin text")`.

Många funktioner kan ta ett eller flera valfria argument, som till exempel beepfunktionen tidigare, den kan man ge en frekvens, längd och volym. Om jag vill spela tonen C (262Hz) i 1 sekund med 80% volym skriver jag `brick.sound.beep(262, 1000, 80)`. Observera att tiden anges i millisekunder, dvs 1 sekund är 1000 millisekunder. Man kan skippa argument bakifrån, då får de defaultvärden. Till exempel blir `brick.sound.beep(262, 1000)` tonen A i en sekund, men med volym 30% för 30% är funktionens fördefinierade defaultvärde. Däremot kan man inte skippa argument framifrån, om jag skulle anropa `brick.sound.beep(1000, 80)` skulle det spelas upp en ton med 1000Hz i 80ms.

En annan praktisk funktion (när man bland annat gör musik) är wait() som väntar en specifik tid, angiven i millisekunder. Den pausar programmet i angiven tid.

≫ Skriv ett program som spelar upp björnen sover, eller annan känd melodi. Sök upp noter på nätet och översätt till frekvenser! Finns gott om info på nätet om vilka frekvenser som motsvarar vilka noter.

36.2 Anropa funktioner med returvärde

En annan skillnad från matematiken är att nästan alltid returnerar f(x) ett numeriskt värde. Men inga av de funktioner (print, beep eller text) har returnerat något. Man använder funktionerna för att få något att hända, eller för att gruppera kod som hör ihop och ska utföra något tillsammans.

Men det finns även funktioner som returnerar ett värde. Till exempel funktionen voltage i klassen battery (som också ligger i brick). Den returnerar hur mycket spänning som finns kvar i batteriet i millivolt. Då kan man använda det värdet för att skriva ut på skärmen med print() eller text() funktionen:

```
brick.display.text(brick.battery.voltage())
```

Vi kommer återkomma till fler funktioner som returnerar värden. Typiskt är det för att läsa av sensorer, som vilken färg en färgsensor ser, eller hur långt avstånd en avståndssensor mäter.

36.3 Skapa egna funktioner

Det går självklart att skapa egna funktioner. Det gör man med nyckelordet `def` följt av funktionens namn. Man använder som sagt funktioner för att gruppera kod som hörs ihop och kanske ska kunna anropas från olika ställen i koden. Vi vill nu skapa en funktion för melodin vi skapade tidigare. Vi döper funktionen till `sleeping_bear`. Python gillar i allmänhet inte de svenska bokstäverna å ä ö så man brukar använda engelska för funktionsnamn, men det går såklart bra att även skriva döpa den till `bjornen_sover`.

Vi skriver

```
def sleeping_bear():
    brick.sound.beep(...)
    brick.sound.beep(...)
    brick.sound.beep(...)
    brick.sound.beep(...)
```

Observera kolonet efter parenteserna på raden som definierar funktionen, det är vanligt att man glömmer det. Observera också att man i python indenterar sin kod, det betyder att man har mellanslag före varje anrop inne i en funktion. På det viset vet python att allting som har mellanslag före sig hör ihop och tillhör funktionen `sleeping_bear`.

Vill man ha någon indata till funktionen kan man skriva det inom parenteserna, men mer om det senare när vi har introducerat variabler.

36.4 Variabler

En variabel är som en postlåda. Man kan stoppa in nått i den och spara för att vid ett senare tillfälle plocka fram det. I många språk måste man berätta vad postlådan ska innehålla, men Python är svagt typat, vilket betyder att alla variabler kan innehålla alla sorters datatyper. Däremot kan det ändå vara bra att hålla koll på vad man jobbar med för sorts datatyp. Några vanliga är heltal, flyttal, strängar, boolska värden och listor.

Heltal är till exempel 1, 5, -4 och 1000.

Flyttal är decimaltal, som 1.5, 5.5 och 7.0, observera att flyttal specificeras genom att använda en punkt, inte huruvida de har en decimaldel. Det betyder att det matematiska heltalet 4 kan uttryckas både som ett heltal i python `4` eller som flyttal `4.0`.

≫ Testa att skriva ut (med print-funktionen lämpligast) vad 5/2 är, eller 5.0/2.0. Det första blir ett heltalsbråk, medan det andra ett flyttalsbråk. Vad händer om täljare eller nämnare har olika datatyp?

Strängar avgränsas med citatstecken och är oftast text, som `"Hej"` eller `"#\%&/("`, men det kan också vara en sträng som innehåller ett tal, som `"5"`, men då behandlas det som symbolen för siffran 5 och inte värdet 5.

Det finns bara två boolska värden, och det är `True` och `False`.

Listor är en samling av värden, de skapar man med hakparenteser och kommatecken mellan listans element. Man skulle till exempel kunna ha en lista över bilmärken: `["Volvo", "BMW", "Fiat"]` Det går också att skapa listor med olika datatyper, till exempel `[5, "Fredrik", 3.14, "Kalle Anka"]`

För att skapa en variabel skriver man dess namn följt av ett likhetstecken:

```
reflected_light = 0.56
```

Ofta kanske man vill spara ett värde från en funktion i en variabel, till exempel spänningen från batteriet som vi läste av förut:

```
current_voltage = brick.battery.voltage()
```

Ibland är det smidigt att kunna konvertera mellan olika datatyper. För att göra om något till ett heltal använder man funktionen `int()` som står för integer, för att göra om något till ett flyttal använder man funktionen `float()`, och så slutligen för att göra om det till en sträng använder man funktionen `str()`.

Förutom att addera flyttal och heltal så används plustecknet också för att konkatenera (är ett ord som kommer från latin och betyder kedja ihop) strängar. Oftast vill man skriva ut en variabels värde och en kort beskrivning av vad det är man printar. Till exempel ”Batterinivå: 8.9 volt". Det kan man göra såhär:

```
print("Batterinivå: " + str(brick.battery.voltage()) + " volt")
```

36.5 Konstanter

Konstanter är en specialform av variabler. En variabel som aldrig ändrar värde kallar man för konstant. I många programmeringsspråk (även Python) skrivs konstanters namn med VERSALER.

De skapas på samma sätt som variabler:

```
PI = 3.14
GRAVITY = 9.8
```

Det finns många konstanter som redan är definierade. Som till exempel olika fördefinierade ljudfiler, de ligger i klassen SoundFile. Till exempel `FANFARE`, `KUNG_FU`, `ELEPHANT_CALL` and `DOG_BARK_1`. Du kan spela upp ljud med funktionen file i klassen sound:

```
brick.sound.file(SoundFile.FANFARE)
```

Några andra vanliga konstanter är A, B, C och D som är portarna för motorerna, samt S1, S2, S3, S4 som är portarna för sensorer. De ligger alla i klassen Port. Mer om dessa när vi pratar om objekt nedan.

36.6 Objekt

Objekt är klasser som har instansierats, det betyder att man kan ta en klass och från den mallen skapa flera instanser (som också kan kallas objekt). Till exempel kan man ta en klass för en motor och skapa flera motor-objekt - ett för varje motor som man anslutit till roboten.

Man skapar objekt genom att anropa en speciell funktion som kallas konstruktor (man kan tänka sig att den funktionen konstruerar objektet). Till exempel finns funktionen/konstruktorn Motor() som skapar ett motor-objekt, och konstruktorn UltrasonicSensor() som skapar en ultraljudssensor. Man sparar objekt i variabler. Precis som heltal eller strängar kan sparas i variabler, så kan även objekt sparas i variabler. Faktum är att heltal och strängar också räknas som objekt i python.

Konstruktorn för motorer och sensorer har en parameter som anger vilken port den är ansluten till:

```
right_motor = Motor(Port.B)
left_motor = Motor(Port.C)
distance_sensor = UltrasonicSensor(Port.S4)
```

Dubbelkolla gärna sladdarna för att se vilken port som motorerna och sensorerna är anslutna till. Det kan variera från robot till robot, även om jag försökt sätta dem lika.

36.7 Motorer

Precis som det finns funktioner som tillhör en speciell klass, så finns det också funktioner som tillhör ett speciellt objekt. För motorer kan man anropa run() som startar motorn, funktionen tar en enda parameter som anger motorns hastighet i grader per sekund.

```
motor = Motor(Port.B)
motor.run(200)
```

≫ Det händer ingenting när ni kör ovanstående kod? Varför? Kan ni ordna det?

Ett annat sätt att lösa ovanstående problem är att använda funktionen `run_time(speed, time)` eller `run_angle(speed, angle)` som kör motorn i ett visst antal millisekunder, respektive en viss vinkel i grader.

Både `run_angle()` och `run_time()` tar två extra valfria argument. Det första argumentet är vad som ska hända efter att motorn snurrat klart, ska den stänga av strömmen helt (`Stop.COAST`) eller ska den aktivt försöka hålla kvar vinkeln (`Stop.HOLD`), det finns också en mittemellan som bromsar motorn (`Stop.BRAKE`). Det andra argumentet är `True` om funktionen ska vänta med att returnera tills den nått sin slutgiltiga vinkel/tid eller `False` om funktionen ska returnera direkt. Det kan vara praktiskt om man ska köra två motorer samtidigt (till exempel två motorer som driver hjulen):

```
right_motor = Motor(Port.B)
left_motor = Motor(Port.C)
right_motor.run_angle(60, 360, Stop.HOLD, False)
left_motor.run_angle(60, 360, Stop.HOLD, True)
print("Motorerna har roterat 360 grader med 60% hastighet")
```

Det finns många fler funktioner på Motor-objektet. Till exempel `speed()` som ger hastigheten som motorn just nu snurrar, `stop()` som stannar motorn, `angle()` som ger nuvarande vinkel på motorn och `reset_angle(angle)` som sätter vinkelgivaren till en specifik vinkel (dvs snurrar inte motorn, men ”nollställer” vinkelgivaren).

≫ Ställ upp ett hinder. T.ex. en tom mjölkförpackning eller ett legotorn. Roboten ska starta ungefär 40cm därifrån riktad mot hindret. Programmera roboten att komma så nära hindret som möjligt utan att välta det.

Att få hög noggrannhet är viktigt i många sammanhang. En industrirobot som ska svetsa ska vara nära objektet men inte välta omkull det. En robot som ska servera kaffe får inte spilla utanför koppen, den måste stå precis på rätt ställe. Ni får här programmera en robot att åka till precis rätt ställe!

≫ Programmera roboten att åka i en fyrkant.

För att kalibrera robotar brukar man programmera dem att åka i en fyrkant. Man kan få ut mycket information från ett så litet program:

- Att roboten inte drar snett när den åker rakt. Skulle roboten åka snett kan motorerna vara olika slitna, eller hjulen ha olika diameter
- Att roboten åker rätt längd på varje raksträcka (och lika långt).
- Att roboten svänger precis 90 grader i varje sväng.

Testa att lägga en sväng och en raksträcka i en funktion så behöver ni inte ändra värdet på flera ställen, utan bara i funktionen. Sen anropas funktionen fyra gånger i rad. Mycket smidigt!

≫ Få roboten att åka en viss bana. Banan kan tejpas upp på golvet. Roboten ska hålla sig inom de båda väggrenarna. Roboten ska vara hårdkodad att följa denna specifika bana. Eftersom roboten inte tar in omgivningen är det viktigt att den startar exakt likadant varje gång, både vinkeln och positionen. Man kan tävla om vem som kommer genom banan snabbast!

Banan behöver inte vara tejpad. Ni kan ställa upp en lego-markör eller en kon som roboten ska ta sig runt och sedan tillbaka till starten. Eller flera koner/bollar som roboten ska köra slalom mellan utan att slå ned någon av konerna/bollarna.

≫ Placera ut lite läskburkar som roboten ska ta tag i och flytta. Det kan t.ex. vara i samma bana som uppgiften ovan. Markera ett fast ställe på banan där burken finns så kan de hårdkoda roboten att åka dit och hämta burken. Markera ett annat ställe på banan där burken ska lämnas. Observera att det blir problem om man använder `run_angle()` på motorn som styr klon och man inte återställer klon mellan två körningar, varför? Använd `run_time()` för att slippa tänka på att återställa klon hela tiden.

Lagerrobotar runt om i världen arbetar såhär. De plockar upp ett objekt och transporterar det från punkt A till punkt B. Det kommer att revolutionera transportindustrin. En taxibil behöver ingen förare om den klarar av att köra mellan olika adresser på egen hand.

≫ Placera ut två läskburkar och programmera roboten att byta plats på dem. T.ex. en fanta och en läskbruk. Roboten måste först flytta fantaburken till en temporär lagringsplats innan de flyttar läskburken till platsen där fantan stod, för att sist åka till fantaburken och flytta den till läskburkens plats. Detta kräver stor noggrannhet och det är viktigt att roboten startar på exakt samma ställe varje gång. Ställ burkarna relativt nära varandra för att roboten inte ska komma helt bort sig när den svänger och kör.

Lagerrobotar måste ibland byta plats på saker för att ha saker som är mest populära närmast. På Clas Ohlsson lagret vet jag att de optimerar så att produkter som ofta tar slut är närmast lastbilarna för att snabbt komma iväg till butik.

36.8 Sensorer

Under avsnittet om objekt så skapade vi också ett UltrasonicSensor objekt. Det har en funktion som heter `distance()` som ger avståndet i millimeter till närmaste hinder framför.

Det finns också många andra sensorer man kan skapa objekt för, till exempel:

- `TouchSensor` som har funktionen `pressed()` som ger `True` om sensorn är intryckt och `False` om den är utsläppt.
- `ColorSensor` som har funktionen `reflection()` som ger reflekterad ljusstyrka i procent där 100 betyder allt är så ljust det kan bli, och 0 är allt kolsvart.
- `GyroSensor` som har funktionen `angle()` som ger gyrots nuvarande vinkel.

≫ Testa att skriva ut värden från sensorerna på skärmen.

36.9 Loopar

Robotar upprepar ofta samma sak om och om igen. T.ex. en industrirobot som ska tillvecka bilar måste göra om samma monteringssteg hundratals gånger. Då används loopar! Loopar är bra för att slippa skriva samma rader kod flera gånger i rad. Som programmerare spar du tid!

Loopar kan också kallas för upprepningar eller repetitioner. Vissa kallar det för snurror eller slingor eftersom programmet snurrar runt och gör samma sak om och om igen.

Python har två sorters loopar, while-loopar och for-loopar. Vi kommer här bara gå igenom while-loopen. Det är den som används mest i legoprogramemring. for-loopen kommer vi titta närmare på i lektion 3.

while loopen upprepar något medan ett villkor är sant. Till exempel medan ett avstånd är större än 20cm eller medan ljusstyrkan som ljussensorn ser är mer än 30%.

```
ljussensor = ColorSensor(Port.S2)
while ljussensor.reflection() < 30:
    print("Ljuset är mindre än 30%")
    wait(500)
    brick.sound.beep(300)
print("Nu var ljuset mer (eller lika med) 30%. Loopen bröts.")
```

Precis som när vi skapade funktioner så måste vi indentera vår kod med mellanslag för att python ska veta vad som tillhör loopen. Allting som vi skriver mellanslag före tillhör loopen.

Man använder det ofta för att vänta på något. Till exempel köra en motor tills man trycker på trycksensorn. Först startar man motorn, sen väntar man på att trycksensorn blir intryckt (med en loop som inte gör något), för att efter att loopen är klar stänga av motorn.

```
motor = Motor(Port.A)
trycksensor = TouchSensor(Port.S1)

motor.run()
while not trycksensor.pressed():
    wait(10)
motor.stop()
```

Observera att inga while-loopar kan vara tomma, då får man ett error. Faktum är att man aldrig kan ha ett tomt block som man indenterar, det ger error även för tomma funktioner. Man brukar lägga in en kort paus i sina loopar för att programmet inte ska låsa sig helt.

≫ Göra samma sak som den tidigare uppgiften när roboten skulle åka fram till en tom mjölkförpackning eller legotorn, fast den här gången med hjälp av en ultraljudssensor. Roboten ska stanna när den har något framför sig.

Ni har programmerat en robot som reagerar på ultraljud. En människa kan inte höra ultraljud, men en fladdermus kan. En fladdermus navigerar med hjälp av ultraljud precis på samma sätt som er robot. Den skickar ut en högfrekvent signal och mäter tiden det tar för ekot att komma tillbaka, med lite matte kan fladdermusen då räkna ut hur långt det är till kvistar och grenar i skogen.

≫ Åka fram tills roboten ser en linje. Samma sak som föregående uppgift, fast man använder en ljussensor för att ge roboten syn. Tejpa upp en linje på vit bakgrund och försök få roboten att åka fram tills den ser linjen. Roboten ska stanna när den ser linjen.

Ljussensorer används överallt i industrin! Det är ett viktigt redskap för att ge robotar syn där kameror är för dyrt. Många robotar reagerar på linjer så att de inte kör fel.

≫ Programmera ett inbrottslarm som väntar på att något ska komma framför US sensorn och sedan tjuter!

≫ Klon fastnar ju när motorn står i "dödläge" och man använder `run_angle()`, men när man använder `run_time()` förlorar man lite tid när klon stängt sig. Bättre att använda en sensor som märker när det är tillräckligt stängt. Stäng klon tills trycksensorn är intryckt.

I stort sett alla mekanismer använder ändlägessensorer för att veta när man kan sluta köra en motor. T.ex. i en 3D skrivare sitter trycksensorer i varje ände av rörelsen för att den inte ska ha sönder sig själv. När trycksensorn trycks in vet 3D skrivaren att den ska sluta flytta skrivarhuvudet. Likadana sensorer sitter föressten i vanliga skrivare också! Och i det mesta som rör sig.

≫ Programmera en robot att åka fram och tillbaka mellan två svarta streck på marken. Blir ungefär som uppgiften då roboten skulle stanna vid en svart linje, men nu ska den byta riktning eller börja backa istället.

Om en robot ska vakta ett industriområde (eller en dörr) behöver den åka fram och tillbaka mellan två markeringar och ha koll på vad som händer. Robotar gör ofta väldigt enformigt arbete som människor inte vill göra.

≫ Räkna linjer och stanna efter att roboten passerat 4 linjer. Använd en variabel som håller koll på hur många linjer som passerats. Denna uppgift är lite svårare än de andra.

≫ Åk i en fyrkant med en loop. Ha en variabel som räknar antalet iterationer (dvs varv genom loopen) och bryt efter fyra iterationer.

Programmerare är lata. Genom att använda loopar slipper man mycket arbete. Det är bättre att programmet automatiskt upprepar saker istället för att man ska skriva upprepad kod.

≫ Hålla sig inom ett upptejpat område och städa rent det. Varje gång roboten ser en linje ska den backa, svänga och köra framåt igen. Puttar ut alla läskburkar från området tillslut.

Kan användas för att städa"ett bord (puttar ned allt på golvet). Kanten på bordet reflekterar inget ljus så det tolkas som en svart tejp av roboten. Gräsklipparrobotar brukar ha en linje/slinga runt hela trädgården som roboten ska hålla sig innanför.

≫ Undvika att köra in i något. Samma som senaste uppgiften fast man väntar på att roboten ska känna av något med ultraljudssensorn istället för ljussensorn.

Detta är på samma sätt som robotdammsugare fungerar. En robotdamsugare krockar i saker och märker det och backar och svänger. Er robot är till och med bättre eftersom den inte behöver slå i någonting, den reagerar innan den krockar i nått!

≫ Följa en linje. Ganska klurig uppgift om man aldrig gjort det förut.

Linjeföljande robotar finns överallt! Det finns tävlingar med dem och de används i industrin. Många robotar guidas av kablar. Vår robotgräsklippare hittar tillbaka till laddstationen med hjälp av en guidekabel som den följer på samma sätt som ni precis gjort. En förarlös bil ser linjerna på vägen och följer dem.

36.10 If-satser

Ibland behöver man läsa av variabler eller sensorer utan att skapa en loop för det, för att slippa upprepa eller vänta på ett värde. Man kanske bara momentant vill undersöka om en knapp är intryckt eller om ett hinder finns framför roboten, eller så vill man kolla flera saker i en loop. Då kan man använda if-satser. De kollar ett villkor och kör ett block av kod om villkoret är sant:

```
if ultraljudssensor.distance() < 800:
    print("Avståndet är mindre än 80cm")
```

Man kan också lägga på en else sats som körs om villkoret är falskt:

```
if trycksensor.pressed():
    brick.sound.file(SoundFile.TOUCH)
    brick.sound.file(SoundFile.DETECTED)
else:
    brick.sound.file(SoundFile.NO)
    brick.sound.file(SoundFile.TOUCH)
```

Man kan också kolla om ett av flera villkor uppfylls, den kommer bara köra det första blocket som är sant:

```
if ljussensor.reflected() < 10:
    print("Kolsvart")
elif ljussensor.reflected() < 40:
    print("Mörkt")
elif ljussensor.reflected() < 60:
    print("Molnigt")
elif ljussensor.reflected() < 90:
    print("Vacker dag")
else:
    print("Strålande solsken")
```

≫ Följ linjen tills roboten hittar en burk. Fånga den och sen fortsätta följa linjen. Robotar på pappersbruk förflyttar stora pappersrullar genom att följa linjer på golvet.

≫ Sluta följa linjen när den kommer till en färgad tejpbit / markering.

Robotar använder sig ofta av något som kalla beacon, vilket betyder markörer på svenska. Dessa använder roboten för att veta var den befinner sig. De kan bestå av t.ex. färgmarkeringar eller saker som sänder ut strålning som roboten reagerar på.

37. Mer Python

4 augusti

Idag ska vi lära oss mer avancerad python-programmering och använda det för att lösa programmeringsproblem.

Om ni redan kan Python behöver ni inte läsa alla texten nedan, men gör ändå uppgifterna som är markerade så här:

≫ Första uppgiften är att få igång er dator och utvecklingsmiljön IDLE.

37.1 Pythonterminalen

När ni startar IDLE får ni upp en Pythonterminal (Python Shell). I den kan man direkt skriva in de instruktioner som man vill att datorn ska utföra.

≫ Börja med att testa att använda Pythonterminalen som en miniräknare. Skriv t.ex. in

```
163+24
```

och se vad resultatet blir.

37.2 Kattis

Kattis (open.kattis.com) är en hemsida med många problem från tidigare programmeringstävlingar. För dessa problem ska man skriva ett program som läser in indata och räknar ut något från det. Skapa ett konto där och sök reda på problemen nedan. ≫ Lös Kattis-problemet **hello**. Det är ett exempelproblem för att se att det funkar att skicka in lösningar. Skapa en pythonfil där ni skriver er lösning och testa att köra den för att se resultatet. Lösningen är på en rad, ganska lik raden nedan. Välj Python3 som programmeringsspråk när ni skickar in lösningen till Kattis.

```
print('Skriv rätt mening här!')
```

När ni skickat in ert program som lösning på ett problem kommer Kattis att köra det programmet med många olika indata och kontrollera att det alltid ger rätt svar.

37.3 Operatorer

Operatorer är tecken eller ord som används för att kombinera och manipulera variabler. Det finns många operatorer i Python. Några av de viktigaste (förutom +, -, *, /, =) finns i tabellen nedan.

Python	Matematik	Beskrivning
`x // y`	$\left\lfloor \frac{x}{y} \right\rfloor$	x delat med y avrundat nedåt
`x % y`	$x (\mathrm{mod}\ y)$	Resten av x vid division med y
`x ** y`	x^y	x upphöjt i y
`x += y`	$x \leftarrow x + y$	Samma som `x = x + y`
`x *= y`	$x \leftarrow x \times y$	Samma som `x = x * y`

Tilldelningsoperatorerna -=, /=, %=, osv följer samma princip.

≫ Testa att ni förstår vad alla operatorerna i tabelen ovan gör genom att använda dem i pythonterminalen.

≫ Vad blir resultatet av `-3//4` och `-3%4`? Gissa innan ni testar.

Testa nu det lite svårare problemet

≫ Kattis problem **pot**

Koden nedan är ett exempel på hur man kan läsa in indata och skriva ut resultatet på detta problem. Använd alla de fyra första operatorerna i tabellen ovan i er lösning.

```
N = int(input()) #Läs in antal tal
X = 0 #Initiera summan till 0
for i in range(N): #Loopa igenom talen i indata
    P = int(input()) #Läs in ett tal
    ??? #Räkna upp summan
print(X) #Skriv ut resultatet
```

37.4 Listor

En lista i Python är en ordnad sekvens av objekt (t.ex. strängar eller nummer). Man kan lägga till nya element (objekt) eller ändra de som redan finns hur mycket man vill. Listor skapas genom hakparenteser, och alla listelementen måste åtskiljas av ett ett kommatecken. Objekten som lagras kan vara av vilken typ som helst (man kan till och med lagra listor inuti listor). Man kommer åt objekten på samma sätt som man kommer åt tecknen i en sträng, med indexering och hakparanteser. Lista ut vad som händer på varje rad i exemplet nedan och hur listan ser ut efter varje rad.

```
ledare = ['Val', 'Ulf', 1]
ledare[2] = 'Sebbi'
ledare += ['Erland']
print(ledare[0])
```

Ni kan se hur den ser ut genom att skriva ut den:

```
print(ledare)
```

Man kan skapa listor med många likadana element så här

```
ettor = [1] * 35
```

För att kolla hur lång en list är kan man använda `len`

```
len(ettor)
```

En tuple liknar en lista, men skapas med parenteser och kan inte modifieras efter att den har skapats.

```
a = (1,4,2)
```

Man kan beräkna summan av elementen i en lista med `sum(list)`, man kan sortera en lista med `list.sort()`, man kan lägga ihop två listor med + och det finns många andra inbyggda funktioner för listor som man kan googla sig till.

Tilldelningar med listor skapar normalt inte en kopia, utan en referens till samma lista. Oavsett vilken lista man modifierar kommer båda listorna att ändras. Vad tror ni att koden nedan kommer skriva ut?

```
a = [1, 2]
b = a
b[0] = 4
print(a)
```

37.5 For-loopar

För att göra något med varje element i en lista kan man använda en for-loop.

```
my_list = [1, 4, 2]
for item in my_list:
    print(item)
```

För att göra en lista som upprepar något n gånger kan man skapa en lista med heltalen från 0 till $n-1$ med kommandot `range(n)`.

```
for i in range(10):
    print(i)
```

≫ Kattis problem **modulo**

```
finns = [False]*42 #Skapa lista som indikerar vilka tal vi har sett
for i in range(10): #Loopa igenom indata
    n = int(input()) #Läs in ett tal
    ? #Gör något här
print(sum(finns)) #Skriv ut antal unika tal
```

≫ Kattis problem **flexible**

```
v = input().split(); #Läs första raden av indata
W = int(v[0]) #Rummets bredd
P = int(v[1]) #Antal valbara väggar
L = [int(x) for x in input().split()] #Väggarnas positioner
```

37.6 If-satser

För att program ska kunna välja att göra olika saker används if-satser. If-satser kräver ett boolskt värde (något som är antingen sant eller falskt). Boolska värden kan skapas genom att skriva `True` eller `False` eller med en jämförelse som i tabellen nedan.

Python	Matematik	Beskrivning
`x == y`	$x = y$	x är lika med y
`x != y`	$x \neq y$	x är inte lika med y
`x > y`	$x > y$	x är större än y
`x >= y`	$x \geq y$	x är större än eller lika med y
`x < y`	$x < y$	x är mindre än y
`x <= y`	$x \leq y$	x är mindre än eller lika med y

En if-sats i python kan se ut så här

```
if 3 > 5:
    print('3 är störst')
else:
    print('5 är störst')
```

Man kan kombinera boolska variabler med operatorerna nedan:

Python	Matematik	Beskrivning
`not a`	$\neg a$	inte a
`x and y`	$x \wedge y$	x och y
`x or y`	$x \vee y$	x eller y

≫ Kattis problem **quadrant**

37.7 While-loopar

En While-loop upprepar några instruktioner så länge vilkor uppfylls. Instruktionerna som ska upprepas är indenterade på samma sätt som för if-satser. Skriv av och kör programmet nedan och försök lista ut hur det fungerar:

```
i = 0
while i < 8:
        print(i)
        i = i + 1
```

Programmet nedan räknar ut summan av talen från 1 till 20.

```
i = 1
summa = 0
while i <= 20:
        summa = summa + i
        i = i + 1
print(summa)
```

≫ Modifiera programmet ovan så att det istället beräknar produkten av heltalen från 5 till 15.

≫ Börja med talet 2017. Addera 42 till det och dividera det sedan med 3. Upprepa detta tills talet blir lägre än 20. Hur många upprepningar tog det?

37.8 Extraproblem

Har ni kommit hit har ni lärt er de viktigaste koncepten för programmering i Python. Det är tillräckligt för att göra i princip vilka beräkningar som helst och lösa i nästan alla problem på Project Euler och Kattis, men det finns många knep kvar som gör det enklare att programmera. En bra mer detaljerad tutorial till python där ni kan hitta många fler knep som gör era program kortare och tydligare är https://docs.python.org/3.6/tutorial/index.html. Google är också bra på att besvara frågor om python (oftast bättre på engelska).

Här är några till problem som ni kan försöka lösa.

- Kattis problem **sumoftheothers**
- Kattis problem **blackfriday**
- Kattis problem **falling**

38. Binärsökning

8 augusti

38.1 Gissa tal

Tänk på ett tal mellan 1 och 100. Med några frågor på formen "Är talet högre än 28?", kan man lista ut vilket tal det är.

- Vilket tal bör man gissa på först för att behöva så få frågor som möjligt?
- Hur fortsätter man när man fått ett svar?
- Hur många frågor behöver man som mest?
- Lös kattisproblemet **Guess the number**. Utgå gärna från koden nedan

```
?? #skapa variabler för att hålla koll på vad talet kan vara som minst och högst
while True:
    g = ? #räkna ut vad vi ska gissa på
    print(g)
    reply = input()
    if reply == "correct": #om det var rätt
        break #avsluta progrmmet
    if reply == "lower": #om det var för lågt
        ? #uppdatera gränserna för vad talet kan vara
    else: #om det var för högt
        ? #uppdatera gränserna för vad talet kan vara
```

38.2 Ekvationslösning

Man kan använda samma strategi för att lösa mycket svåra ekvationer numeriskt. Man börjar då med att skapa en funktion $f(x)$ så att ekvationen blir $f(x) = 0$.

≫ Vad blir $f(x)$ om man vill lösa ekvationen $xe^x = 50$? Implementera denna funktion i python och beräkna $f(5)$. För att skriva e^x i python behöver man importera modulen `math` och kan sedan skriva `math.exp(x)`.

För att lösa ekvationen måste man hitta en under och en övre gräns för vad lösningen kan vara. Man behöver alltså ett x_1 så att $f(x_1) < 0$ och ett x_2 så att $f(x_2) > 0$.

≫ Hitta något x_1 och x_2 som uppfyller villkoren ovan.

≫ Skriv ett python-program som löser ekvationen. Ni kan utgå från er lösning på kattisproblemet Guess the number om ni vill. Avbryt när intervallet ni begränsat svaret till är mindre än 10^{-8}.

≫ Hur många iterationer behöver loopen göra för att hitta svaret? Kan ni räkna ut antal iteration utifrån x_1, x_2 och det maximala felet

≫ Försök nu lösa ekvationen $\frac{x^2}{\sin(x)} = 2 - x$

38.3 Extrauppgifter

Lös någon av uppgifterna nedan om ni har tid över

≫ Kattis problem **carpet**

≫ Kattis problem **billiard**

≫ Kattis problem **bookingaroom**

39. Giriga algoritmet

10 augusti

Giriga algoritmer är algoritmer som löser ett större problem i små steg genom att hela tiden göra det som verkar bäst för tillfället.

39.1 Mynt

Exempel: Antag att ni har mynt med värdena 1 kr, 2 kr, 5 kr och 10 kr. Vad är det minsta antalet mynt som krävs för att få 79 kr?

Det kommer gå åt färre mynt om vi använder mynt med högre värden. Vi vill därför använda så många mynt med höga värden som möjligt. Vi kan börja med att räkna ut hur många 10-kronor vi som mest kan använda. Efter det kan vi räkna ut hur många 5-kronor vi kan använda för värdet som är kvar, osv med 2- och 1-kronorna.

Detta är en girig algoritm eftersom den bara tittar på en typ av mynt i taget utan att ta hänsyn till hur det kommer lösa sig med det resterande värdet och de mynt som är kvar. Giriga algoritmer fungerar inte på alla problem. Om mynten hade haft värdena 1, 4 och 5 kr istället så hade algoritmen ovan inte säkert gett rätt svar. Om vi skulle räknat upp summan 8 kr hade den giriga algoritmen gett en femma och tre enkronor, när det hade blivit färre mynt med två 4-kronor.

≫ Skriv ett program som tar som indata summan vi ska få och svarar hur många 1-kronor, 2-kronor, 5-kronor och 10-kronor vi ska använda för att få denna summa med så få mynt som möjligt. Ni kan börja med koden nedan:

```
V = int(input()) #Läs in totala värdet
n10 = V//10 #beräkna antal 10-kronor
V -= n10*10 #ta bort värdet av 10-kronorna
n5 = ...
...
print(n1, 'enkronor,', n2, 'tvåkronor,', n5, 'femmor, och', n10, 'tior')
```

39.2 Fler problem med giriga lösningar

Försök lösa vad ni hinner av problemen nedan

≫ Kattis problem **color**

≫ Kattis problem **virus**

≫ Kattis problem **a1paper**

≫ Kattis problem **zigzag**

V Lektionsmaterial programmeringsgruppen Lovelace

40. Legorobotar

2 augusti

Idag ska vi installera utvecklingsmiljön för att programmera Legorobotar med Python. Ni kanske har programmerat legorobotar tidigare med grafiska block, men man kan faktiskt programmera dem med Python och andra textuella programmeringsspråk också. Vi kommer använda oss av programmet Visual Studio Code, med ett tillägg för EV3 Mindstorms.

Vi förutsätter att ni har programmerat tidigare och går mest igenom vad som är specifikt för Python. Vi jämför också en del med andra programmeringsspråk.

40.1 Imports

Det första ni ser när ni skapar ett EV3 Mindstorms program i Visual Studio Code är ett gäng imports. Det är rader som importerar kod och funktionalitet från andra filer, precis som #include i C och C++ eller require i JavaScript. Vi kan antingen importer ett helt bibliotek, till exempel

```
import math
```

och då kommer vi åt variabler och funktioner genom att skriva `math.` följt av variabel eller funktionsnamn. Till exempel:

```
import math
print(math.sin(2 * math.pi))
```

Ett sätt att slippa skriva math framför är att välja vilka funktioner eller variabler man vill inkludera:

```
from math import pi, sin
print(sin(2 * pi))
```

Vet man inte vilka funktioner man vill använda, eller vill använda så många så man inte orkar skriva ut namnet på alla så kan man importera allting genom att skriva stjärna *, men det är dålig praxis:

```
from math import *
print(sin(2 * pi))
```

≫ Blir svaret 0 på ovanstående? Varför eller varför inte?

40.2 Funktioner

Python är objektorienterat och baseras på klasser. Till exempel ligger alla funktioner som har något med ev3-klossen att göra i klassen brick. Sen finns det underklasser till den, till exempel allting som har med högtalaren att göra ligger i en klass som heter sound. Så för att spela upp ett ljud kan man skriva `brick.sound.beep()`.

Det går också att skriva ut saker på ev3-enhetens skärm, då använder man funktionen text() i klassen display som ligger i brick-klassen. För att skriva ut något skriver man `brick.display.text("En fin text")`.

Många funktioner kan ta ett eller flera valfria argument, som till exempel beep-funktionen tidigare, den kan man ge en frekvens, längd och volum. Om jag vill spela tonen C (262Hz) i 1 sekund med 80% volym skriver jag `brick.sound.beep(262, 1000, 80)`. Observera att tiden anges i millisekunder, dvs 1 sekund är 1000 millisekunder. Man kan skippa argument bakifrån, då får de defaultvärden. Till exempel blir `brick.sound.beep(262, 1000)` tonen A i en sekund, men med volym 30% för 30% är funktionens fördefinierade defaultvärde. Däremot kan man inte skippa argument framifrån, om jag skulle anropa `brick.sound.beep(1000, 80)` skulle det spelas upp en ton med 1000Hz i 80ms.

En annan praktisk funktion (när man bland annat gör musik) är wait() som väntar en specifik tid, angiven i millisekunder.

≫ Skriv ett program som spelar upp björnen sover, eller annan känd melodi. Sök upp noter på nätet och översätt till frekvenser! Finns gott om info på nätet om vilka frekvenser som motsvarar vilka noter.

40.3 Skapa egna funktioner

Det går självklart att skapa egna funktioner. Det gör man med nyckelordet `def` följt av funktionens namn. Vi vill nu skapa en funktion för melodin vi skapade tidigare. Vi skriver:

```
def sleeping_bear():
    brick.sound.beep(...)
    brick.sound.beep(...)
    brick.sound.beep(...)
    brick.sound.beep(...)
```

Observera kolonet efter parenteserna på raden som definierar funktionen, det är vanligt att man glömmer det. Observera också att man i python indenterar sin kod, det betyder att man har mellanslag före varje anrop inne i en funktion. På det viset vet python att allting som har mellanslag före sig hör ihop och tillhör funktionen `sleeping_bear`.

Eventuella inargument till funktionen skrivs mellan parenteserna, separerade med komma. Vill man lägga till default-argument så skriver man dem med likhetstecken i argumentlistan:

```
def drive_robot(distance, speed = 40):
    # Kör roboten en specifik sträcka. Om man inte anger hastighet blir den 40.
    pass
```

40.4 Variabler och datatyper

I många språk måste man deklarera de variabler man ska använda och deras datatyp, men Python är svagt typat och du behöver inte deklarera någon variabel.

Det går att använda variabler direkt, och man kan byta vad för datatyp som lagras i variabeln.

```
en_variabel = "hej"
en_variabel = 3.14
```

Däremot finns det ändå olika datatyper i python. Några vanliga är heltal, flyttal, komplexa tal, strängar, boolska värden och listor. Flyttal specificeras genom att använda en punkt, inte huruvida de har en decimaldel. Det betyder att det matematiska heltalet 4 kan uttryckas både som ett heltal i python `4` eller som flyttal `4.0`. Man kan också skriva det som ett flyttal i grundpotensform som 4e0 (det är inte talet e som används här, utan e är en förkortning av ordet exponent). Komplexa tal anges med j som den komplexa enheten: `5+4j`.

Heltal kan man specificera i olika talbaser:

```
a = 0b1010 #Binary Literals
b = 100    #Decimal Literal
c = 0o310  #Octal Literal
d = 0x12c  #Hexadecimal Literal
```

≫ Testa att skriva ut (med print-funktionen lämpligast) vad 5/2 är, eller 5.0/2.0. Det första blir ett heltalsbråk, medan det andra ett flyttalsbråk. Vad händer om täljare eller nämnare har olika datatyp?

Strängar avgränsas med enkla eller dubbla citatstecken.

Listor eller Arrays skapar man med hakparenteser och kommatecken mellan listans element. Det går också att skapa listor med olika datatyper, till exempel `[5, "Fredrik", 3.14, "Kalle Anka"]`

Ibland är det smidigt att kunna konvertera mellan olika datatyper. För att göra om något till ett heltal använder man funktionen `int()` som står för integer, för att göra om något till ett flyttal använder man funktionen `float()`, och så slutligen för att göra om det till en sträng använder man funktionen `str()`.

Förutom att addera flyttal och heltal så används plustecknet också för att konkatenera (är ett ord som kommer från latin och betyder kedja ihop) strängar. Oftast vill man skriva ut en variabels värde och en kort beskrivning av vad det är man printar. Till exempel ”Batterinivå: 8.9 volt". Det kan man göra såhär:

```
print("Batterinivå: " + str(brick.battery.voltage()) + " volt")
```

≫ Vad händer om man skriver `print("5" + 4)` eller `print("5" * 4)`? Testa båda!

40.5 Konstanter

I många programmeringsspråk (även Python) skrivs konstanters namn med VERSALER. Faktiskt så är det inte riktiga konstanter, för det finns inget *const* nyckelord i Python, men av konvention skriver man aldrig över värdet i en variabel med versalt namn.

De skapas på samma sätt som variabler:

```
PI = 3.14
GRAVITY = 9.8
```

Det finns många konstanter som redan är definierade. Som till exempel olika fördefinierade ljudfiler, de ligger i klassen SoundFile. Till exempel `FANFARE`, `KUNG_FU`, `ELEPHANT_CALL` and `DOG_BARK_1`. Du kan spela upp ljud med funktionen file i klassen sound:

```
brick.sound.file(SoundFile.FANFARE)
```

Några andra vanliga konstanter är A, B, C och D som är portarna för motorerna, samt S1, S2, S3, S4 som är portarna för sensorer. De ligger alla i klassen Port. Mer om dessa när vi pratar om objekt nedan.

40.6 Objekt

Objekt är klasser som har instansierats, det betyder att man kan ta en klass och från den mallen skapa flera instanser (som också kan kallas objekt). Till exempel kan man ta en klass för en motor och skapa flera motor-objekt - ett för varje motor som man anslutit till roboten.

Konstruktorn för motorer och sensorer har en parameter som anger vilken port den är ansluten till:

```
right_motor = Motor(Port.B)
left_motor = Motor(Port.C)
distance_sensor = UltrasonicSensor(Port.S4)
```

Dubbelkolla gärna sladdarna för att se vilken port som motorerna och sensorerna är anslutna till. Det kan variera från robot till robot, även om jag försökt sätta dem lika.

40.7 Motorer

I python kallas funktioner som inte tillhör något objekt för funktioner, medan funktioner som tillhör ett instaniserat objekt kallas metoder. Både funktioner och metoder anropas på samma sätt. På motor-objektet finns metoden `run()` som startar motorn, funktionen tar en enda parameter som anger motorns hastighet i grader per sekund.

```
motor = Motor(Port.B)
motor.run(200)
```

≫ Det händer ingenting när ni kör ovanstående kod? Varför? Kan ni ordna det?

Ett annat sätt att lösa ovanstående problem är att använda funktionen `run_time(speed, time)` eller `run_angle(speed, angle)` som kör motorn i ett visst antal millisekunder, respektive en viss vinkel i grader.

Både `run_angle()` och `run_time()` tar två extra valfria argument. Det första argumentet är vad som ska hända efter att motorn snurrat klart, ska den stänga av strömmen helt (`Stop.COAST`) eller ska den aktivt försöka hålla kvar vinkeln (`Stop.HOLD`), det finns också en mittemellan som bromsar motorn (`Stop.BRAKE`). Det andra argumentet är `True` om funktionen ska vänta med att returnera tills den nått sin slutgiltiga vinkel/tid eller `False` om funktionen ska returnera direkt. Det kan vara praktiskt om man ska köra två motorer samtidigt (till exempel två motorer som driver hjulen):

```
right_motor = Motor(Port.B)
left_motor = Motor(Port.C)
right_motor.run_angle(60, 360, Stop.HOLD, False)
left_motor.run_angle(60, 360, Stop.HOLD, True)
print("Motorerna har roterat 360 grader med 60% hastighet")
```

Det finns många fler funktioner på Motor-objektet. Till exempel `speed()` som ger hastigheten som motorn just nu snurrar, `stop()` som stannar motorn, `angle()` som ger nuvarande vinkel på motorn och `reset_angle(angle)` som sätter vinkelgivaren till en specifik vinkel (dvs snurrar inte motorn, men "nollställer" vinkelgivaren).

För att underlätta för oss att styra en robot med två hjul (som den ni har fått här) finns det en klass som heter DriveBase vars konstruktor tar in två motorer, hjulens diameter och avståndet mellan hjulens mittpunkter. Den kan användas för att styra roboten smidigare, utan att styra varje enskild motor var för sig:

```
left = Motor(Port.B)
right = Motor(Port.C)
robot = DriveBase(left, right, 56, 114)

# Drive forward at 100 mm/s for two seconds
robot.drive(100, 0, 2000)
# Turn at 45 deg/s for three seconds
robot.drive(0, 45, 3000)
```

≫ Programmera roboten att åka i en fyrkant.

För att kalibrera robotar brukar man programmera dem att åka i en fyrkant. Man kan få ut mycket information från ett så litet program:

- Att roboten inte drar snett när den åker rakt. Skulle roboten åka snett kan motorerna vara olika slitna, eller hjulen ha olika diameter
- Att roboten åker rätt längd på varje raksträcka (och lika långt).
- Att roboten svänger precis 90 grader i varje sväng.

Testa att lägga en sväng och en raksträcka i en funktion så behöver ni inte ändra värdet på flera ställen, utan bara i funktionen. Sen anropas funktionen fyra gånger i rad. Mycket smidigt!

≫ Få roboten att åka en viss bana. Banan kan tejpas upp på golvet. Roboten ska hålla sig inom de båda väggrenarna. Roboten ska vara hårdkodad att följa denna specifika bana. Eftersom roboten inte tar in omgivningen är det viktigt att den startar exakt likadant varje gång, både vinkeln och positionen. Man kan tävla om vem som kommer genom banan snabbast!

Banan behöver inte vara tejpad. Ni kan ställa upp en lego-bit eller en kon som roboten ska ta sig runt och sedan tillbaka till starten. Eller flera koner/bollar som roboten ska köra slalom mellan utan att slå ned någon av konerna/bollarna.

≫ Placera ut två läskburkar och programmera roboten att byta plats på dem. T.ex. en fanta och en läskbruk. Roboten måste först flytta fantaburken till en temporär lagringsplats innan de flyttar läskburken till platsen där fantan stod, för att sist åka till fantaburken och flytta den till läskburkens plats. Detta kräver stor noggrannhet och det är viktigt att roboten startar på exakt samma ställe varje gång. Ställ burkarna relativt nära varandra för att roboten inte ska komma helt bort sig när den svänger och kör.

Lagerrobotar måste ibland byta plats på saker för att ha saker som är mest populära närmast. På Clas Ohlsson lagret vet jag att de optimerar så att produkter som ofta tar slut är närmast lastbilarna för att snabbt komma iväg till butik.

40.8 Sensorer

Under avsnittet om objekt så skapade vi också ett UltrasonicSensor objekt. Det har en funktion som heter `distance()` som ger avståndet i millimeter till närmaste hinder framför.

Det finns också många andra sensorer man kan skapa objekt för, till exempel:

- `TouchSensor` som har funktionen `pressed()` som ger `True` om sensorn är intryckt och `False` om den är utsläppt.
- `ColorSensor` som har funktionen `reflection()` som ger reflekterad ljusstyrka i procent där 100 betyder allt är så ljust det kan bli, och 0 är allt kolsvart.
- `GyroSensor` som har funktionen `angle()` som ger gyrots nuvarande vinkel.

≫ Testa att skriva ut värden från sensorerna på skärmen.

40.9 Loopar

Python har två sorters loopar, while-loopar och for-loopar. Jag skulle vilja påstå att while-loopen är den vanligaste när man programmerar legorobotar.

while loopen upprepar något medan ett villkor är sant. Till exempel medan ett avstånd är större än 20cm eller medan ljusstyrkan som ljussensorn ser är mer än 30%.

```
ljussensor = ColorSensor(Port.S2)
while ljussensor.reflection() < 30:
    print("Ljuset är mindre än 30%")
    wait(500)
    brick.sound.beep(300)
print("Nu var ljuset mer (eller lika med) 30%. Loopen bröts.")
```

Precis som när vi skapade funktioner så måste vi indentera vår kod med mellanslag för att python ska veta vad som tillhör loopen. Allting som vi skriver mellanslag före tillhör loopen.

Man använder det ofta för att vänta på något. Till exempel köra en motor tills man trycker på trycksensorn. Först startar man motorn, sen väntar man på att trycksensorn blir intryckt (med en loop som inte gör något), för att efter att loopen är klar stänga av motorn.

```
motor = Motor(Port.A)
trycksensor = TouchSensor(Port.S1)

motor.run()
while not trycksensor.pressed():
    wait(10)
motor.stop()
```

Observera att inga while-loopar kan vara tomma, då får man ett error. Faktum är att man aldrig kan ha ett tomt block som man indenterar, det ger error även för tomma funktioner. Man brukar lägga in en kort paus i sina loopar för att programmet inte ska låsa sig helt. Ett annat vanligt knep är att lägga in en NOP-instruktion, NOP står för No-Operation och är en tom instruktion som inte gör något. I python kallas den pass. Så ovanstående loop kan skrivas såhär:

```
while not trycksensor.pressed():
    pass
```

Men jag rekommenderar en liten kort wait / sleep, för att låta processorn hinna bearbeta andra saker i bakgrunden.

≫ Åka fram tills roboten ser en linje. Tejpa upp en linje på vit bakgrund och försök få roboten att åka fram tills den ser linjen. Roboten ska stanna när den ser linjen.

Ljussensorer används överallt i industrin! Det är ett viktigt redskap för att ge robotar syn där kameror är för dyrt. Många robotar reagerar på linjer så att de inte kör fel.

≫ Programmera ett inbrottslarm som väntar på att något ska komma framför US sensorn och sedan tjuter!

≫ Klon fastnar ju när motorn står i "dödläge" och man använder `run_angle()`, men när man använder `run_time()` förlorar man lite tid när klon stängt sig. Bättre att använda en sensor som märker när det är tillräckligt stängt. Stäng klon tills trycksensorn är intryckt.

I stort sett alla mekanismer använder ändlägessensorer för att veta när man kan sluta köra en motor. T.ex. i en 3D skrivare sitter trycksensorer i varje ände av rörelsen för att den inte ska ha sönder sig själv. När trycksensorn trycks in vet 3D skrivaren att den ska sluta flytta skrivarhuvudet. Likadana sensorer sitter föressten i vanliga skrivare också! Och i det mesta som rör sig.

≫ Programmera en robot att åka fram och tillbaka mellan två svarta streck på marken. Blir ungefär som uppgiften då roboten skulle stanna vid en svart linje, men nu ska den byta riktning eller börja backa istället.

Om en robot ska vakta ett industriområde (eller en dörr) behöver den åka fram och tillbaka mellan två markeringar och ha koll på vad som händer. Robotar gör ofta väldigt enformigt arbete som människor inte vill göra.

≫ Räkna linjer och stanna efter att roboten passerat 4 linjer. Använd en variabel som håller koll på hur många linjer som passerats. Denna uppgift är lite svårare än de andra.

≫ Åk i en fyrkant med en loop. Ha en variabel som räknar antalet iterationer (dvs varv genom loopen) och bryt efter fyra iterationer.

Programmerare är lata. Genom att använda loopar slipper man mycket arbete. Det är bättre att programmet automatiskt upprepar saker istället för att man ska skriva upprepad kod.

≫ Hålla sig inom ett upptejpat område och städa rent det. Varje gång roboten ser en linje ska den backa, svänga och köra framåt igen. Puttar ut alla läskburkar från området tillslut.

Kan användas för att städa"ett bord (puttar ned allt på golvet). Kanten på bordet reflekterar inget ljus så det tolkas som en svart tejp av roboten. Gräsklipparrobotar brukar ha en linje/slinga runt hela trädgården som roboten ska hålla sig innanför.

≫ Undvika att köra in i något. Samma som senaste uppgiften fast man väntar på att roboten ska känna av något med ultraljudssensorn istället för ljussensorn.

Detta är på samma sätt som robotdammsugare fungerar. En robotdamsugare krockar i saker och märker det och backar och svänger. Er robot är till och med bättre eftersom den inte behöver slå i någonting, den reagerar innan den krockar i nått!

≫ Följa en linje. Ganska klurig uppgift om man aldrig gjort det förut.

Linjeföljande robotar finns överallt! Det finns tävlingar med dem och de används i industrin. Många robotar guidas av kablar. Vår robotgräsklippare hittar tillbaka till laddstationen med hjälp av en guidekabel som den följer på samma sätt som ni precis gjort. En förarlös bil ser linjerna på vägen och följer dem.

40.10 If-satser

Även python har ifsatser. Notera att de slutar med kolon och att man inte har villkoret inom parenteser (som i t.ex. Java eller C / C++). Även här måste koden som tillhör ifsatsen indenteras korrekt:

```
if ultraljudssensor.distance() < 800:
    print("Avståndet är mindre än 80cm")
```

Man kan också lägga på en else sats som körs om villkoret är falskt:

```
if trycksensor.pressed():
    brick.sound.file(SoundFile.TOUCH)
    brick.sound.file(SoundFile.DETECTED)
else:
    brick.sound.file(SoundFile.NO)
    brick.sound.file(SoundFile.TOUCH)
```

Man kan också kolla om ett av flera villkor uppfylls, den kommer bara köra det första blocket som är sant:

```
if ljussensor.reflected() < 10:
    print("Kolsvart")
elif ljussensor.reflected() < 40:
    print("Mörkt")
```

```
elif ljussensor.reflected() < 60:
    print("Molnigt")
elif ljussensor.reflected() < 90:
    print("Vacker dag")
else:
    print("Strålande solsken")
```

≫ Följ linjen tills roboten hittar en burk. Fånga den och sen fortsätta följa linjen. Robotar på pappersbruk förflyttar stora pappersrullar genom att följa linjer på golvet.

≫ Sluta följa linjen när den kommer till en färgad tejpbit / markering.

Robotar använder sig ofta av något som kalla beacon, vilket betyder markörer på svenska. Dessa använder roboten för att veta var den befinner sig. De kan bestå av t.ex. färgmarkeringar eller saker som sänder ut strålning som roboten reagerar på.

41. Pygame

3 augusti

Idag ska vi lära oss att göra grafik och spel i pygame.

Börja med att installera python-modulen pygame t.ex. med kommandot

```
python3 -m pip install -U pygame.
```

41.1 Luffarschack

Luffarschack är en enkelt men komplext spel som man kan spela med papper och penna, men här kommer vi implementera det som ett datorspel som man styr med musen.

≫ Ladda ned luffarschack_start.py från `tinyurl.com/y3vkor88` och se att det funkar på din dator. Gör sedan följande förbättringar till spelet:

- Byt ut markörerna mot kryss och cirklar. Googla på pygame circleför att se hur det funkar
- Hindra spelaren från att markera en ruta som redan är tagen
- Kontrollera om någon har vunnit och skriv i så fall ut vem
- Markera ut vilka fem markeringar som gjorde vinsten

- Gör så att man kan vara tre spelare och vinner på 4 i rad

41.2 Fisk

Detta är ett enkelt spel som styrs med tangentbordet.
≫ Ladda ned fish_start.py, fish.png och shark.png från `tinyurl.com/y3vkor88` och se att det funkar på din dator. Gör sedan följande förbättringar:

- Lägg till något för fisken att äta, låt det komma mer mat eftersom man äter upp den gamla
- Ha en räknare som visar hur mycket fisken har ätit
- Skriv ut Game Over! på skärmen om man dör
- Lägg till fler hajar. Om de stöter ihop med varandra dör en eller båda och det skapas nya
- Gör så att man kommer till en svårare bana om man klarar den första

42. Geometri

4 augusti

Idag ska vi börja med algoritmisk problemlösning med fokus på geometri. Vi kommer träna detta med problem från gamla programmeringstävlingar.

Vi kommer börja prata om några vanliga subproblem i geometriproblem.

- Representation av punkter och linjer
- Avstånd mellan punkter
- Beräkning av area på en triangel
- Hur man beräknar arean på en godtycklig polygon
- Hur man avgör vilken sida om en linje en punkt ligger
- Beräkning av konvext hölje

42.1 Kattis

Kattis (open.kattis.com) är en hemsida med många problem från tidigare programmeringstävlingar. För dessa problem ska man skriva ett program som läser in indata och räknar ut något från det. Skapa ett konto där och sök reda på problemen nedan. Hello är bara ett testproblem så ni ser att ni förstått hur man skickar in ett problem. Var noga med att välja Python3 som programmeringsspråk. När ni skickat in ert program som lösning på ett problem kommer kattis att köra det programmet med många olika indata och kontrollera att det alltid ger rätt svar.

≫ Kattis problem **hello**

42.2 Toast

Vi börjar med ett matematiskt klurigt men programmeringsmässingt enkelt problem ≫ Kattis problem **toast**

N people are sitting evenly spaced around a circular table. All of them are infinitesimally small, indeed they can be modeled as points, except that they all have pretty long arms - *D* cm long arms, to be exact. Amer pronounces a toast, and everyone cheers! Well, everyone clinks their glass with everyone that they can reach. In other words, two people will clink glasses if their arms can reach each other across the table. In total, you hear *T* “clink!” sounds as the milk glasses touch. What is the radius of the table?

Rita en figur och skriv ned ekvationerna som relaterar N, D och T till varandra och radien på bordet. Nedan finns koden för inläsning av indata ut utskrift av resultatet men ni behöver lägga in beräkningarna själva.

```
import math #ger math.pi och math.sin()

v = input().split() #Läs in raden med indata
N = int(v[0]) #Antal personer
D = int(v[1]) #Armlängden
T = int(v[2]) #Antal "clink!"

n = T/N #Antal personer man når åt höger
if ?: #Specialfall om alla når alla
    l = ? #Radien på minsta möjliga bordet
    h = ? #Radien på största möjliga bordet
else:
    ??? #Beräkna l och h på något sätt
print(l,h) #Skriv ut resultatet
```

42.3 Flower garden

För detta problem krävs mer programmering och mindra matematik.

≫ Kattis problem **flowergarden**

Due to the high cost of flowers in Alberta, Yraglac decided it would be a good idea to plant his own flower garden instead of buying flowers for his girlfriend. We can model his flower garden on a two-dimensional grid, where the position of each flower is given by an (x,y) coordinate. Today, Yraglac decided it would be a good idea to water his flowers one by one, starting at the origin (0,0) and walking to each flower. He will take the shortest path at each step. However, it turns out that Yraglac has an important meeting today and has a limited amount of time available to water the flowers. Thus, he has ranked all of the flowers in his garden from most to least important and will water them in that order. For religious reasons, the number of flowers watered by Yraglac must be a prime number, or zero. Help Yraglac find the maximum number of flowers he can water.

```
import math #Ger math.sqrt för roten ur

def is_prime(p): #Avgör om ett tal är ett primtal
    if p == 1:
        return False
    for i in range(2, min(p, round(math.sqrt(p))+1)):
        if p%i == 0:
            return False
    return True

T = int(input()) #Antal testfall
for i in range(T): #Loopa igenom testfallen
    v = input().split() #Rad med indata
    N = int(v[0]) #Antal blommor
    D = int(v[1]) #Maximal totalsträcka
    x = [0]*N #x-koordinater för blommor
    y = [0]*N #y-koordinater för blommor
    for j in range(N): #Läs in alla blomkoordinater
        v = input().split()
        x[j] = int(v[0])
        y[j] = int(v[1])

    dist = 0 #Hur långt vi gått hittils
    watered = 0 #Hur många blommor vi vattnat hittils
    x = [0] + x #Lägg till startpositionen i koordinatlistan
    y = [0] + y
    for j in range(1,N+1): #Gå mellan blommorna
        ??

    print(watered) #Skriv ut resultatet för detta testfall
```

42.4 Fler geometriproblem

När ni blir klara med problemen ovan kan ni fortsätta så långt ni hinner med problemen nedan.

≫ Kattis problem **galactic**
(Klurig, men kort implementation)

≫ Kattis problem **jabuke**
(Triangelarea och punkter inom triangel)

≫ Kattis problem **roundedbuttons**
(Ganska enkel)

≫ Kattis problem **simplepolygon**
(lite klurig)

≫ Kattis problem **tomosynthesis**
(lite matte och lite programmering)

≫ Kattis problem **walls**
(testa olika fall)

≫ Kattis problem **billiard**
(kom på knepet så blir den lätt)

≫ Kattis problem **convexhull**
(svår)

≫ Kattis problem **wrapping**
(kräver convex hull)

43. Genetiska algoritmer

5 augusti

Genetiska algoritmer är en klass av algoritmer som kan användas för att ssmart slumpafram lösningar till många olika sorters problem. För det mesta är de sämmre än mer specialiserad algoritmer, men har som fördel att de kan appliceras på en väldigt stor mängd väldigt olika problem.

Tanken är att försöka härma evolutionen (därav namnet) genom att producera flera serier med generationer, där de bäst lämpade överlever och förökar sig medan de sämsta dör och försvinner. Gör man detta tillräckligt länge så hoppas vi att vi får en lösning som i slutet är bra nog för våra behov.

Video: https://www.youtube.com/watch?v=pgaEE27nsQw

Problem 1. Skriv en genetisk algoritm som kan approximera en lösning till en ekvationen $2^x = 50$.

(Om du vill kan du istället använda någon annan ekvation, t.ex. $x^3 - 1472x^2 - 574x + 3634 = 0$, eller $\Sigma_{i=0}^{10} 2^{ix} = 10$. Försök skriva din kod så att det blir enkelt att byta ut ekvationen!)

1. Börja med att skriva en funktion `fel(x)` som kan testa en lösning. Den bör ta ett tal `x` och returnera vad skillnaden mellan höger- och vänsterledet blir om man sätter $x =$ `x`

2. Nu ska vi skriva en funktion `para(mor, far)` som kombinerar två lösningar till en (förhoppningsvis!) bättre ny lösning. Enklast är att ta två tidigare lösningar (kallade `mor` och `far`) och låta `barn` vara medelvärdet mellan `mor` och `far`

3. Nästa steg är skapa en ny generation från en gammal, i funktionen `nygeneration(gammalgene`
   Det är viktigt att den nya generationen har lika många lösningar som den gamla, så att populationsstorleken inte växer eller krymper.

   (a) Börja med att sortera `gammalgeneration` så att de lösningar med minst `fel` ligger först.

   (b) Skapa sen en ny tom list, kallad `nygeneration`

   (c) Ta de 100 bästa lösningarna från `gammalgeneration` och lägg dem i `nygeneration`

   (d) Skapa sen 900 nya lösningar i `nygeneration` genom att använda funktionen `para`. Slumpa fram `mor` och `far` från de 500 första lösningarna i `gammalgeneration`

4. Slumpa fram en lista `losningar` med 1000 möjliga lösningar. Alla dessa tal bör vara mellan 0 och 100, inklusive. (Du kan använda modulen `random` för att generera slumptal.) Försök se till att få med inte bara heltal utan också några decimaltal.

5. Skriv nu en loop där man 10000 gånger använder funktionen `nygeneration` med resultatet från det förra varvet. (Första varvet får ni använda listan `losningar`.)

6. Skriv nu kod som efter loopen tar och sorterar den sista generationen, och skriv ut vilken den bästa lösningen ni har hittat är.

7. Kör programmet ni nu har skrivit. Använd en miniräknare eller dyl. för att kontrollera hur bra ert svar är!

Det program ni har skrivit nu ger förhoppningsvis ett relativt bra svar, men har ett problem:

Fråga 1. Anta att det slumpar sig så att det minsta talet som slumpas fram i steg 4. är 2 och det största talet är 97. Kan ni då säga något om hur stora eller små det slutgiltiga svaret kan vara?

Förbättring 1.För att lösa detta problem använder vi *mutation*.

1. Skriv en funktion `mutera(barn)` som slumpmässigt väljer ett tal `mutation` mellan 0.9 och 1.1 och returnerar `barn * mutation`

2. Modifiera funktionen `para` så att den har 0.1% chans att mutera barnet med hjälp av funktionen `mutera`

3. Kör programmet igen och se vad ni nu får för svar. Jämför hur bra det är med det svar ni fick förra gången.

Genom att använda mutation så introducerar man slump inte bara i början av programmet när man skapar `losningar`, utan också lite hela tiden genom programmets körning. På det sättet jämnar man ut slumpens påverkan över alla generationerna.

Det finns många andra sätt att skriva en genetisk algoritm. T.ex. kan man ändra i funktionen `para` så att den bästa lösningen parar sig fler gånger än de sämre, eller hur man nu vill. Experimentera gärna - fundera på hur ni tror att det funkar på riktigt i naturen! Får verkligen de 70% sämsta inte para sig alls där?

Här är två andra problem ni kan försöka lösa med genetiska algoritmer. Ni kan förhoppningsvis återanvända mycket av er kod, men en del måste såklart förändras.

Problem 2. Skriv en algoritm som kan gissa sig till en viss sträng

1. Välja och definiera den "korrektasträngen `svaret`, t.ex. `svaret = "hej valentina!"`

2. Definiera vilka tecken som är okej att ha i strängen, det s.k. alfabetet. T.ex. `alfabetet = "abcdefghijklmnopqrstuvwxyzåäö .!?"`

3. Modifiera koden så att `losningar` istället fylls med strängar av samma längd som `svaret` med tecken tagna från `alfabetet`

4. Modifiera `fel(x)` så att den istället mäter hur långt från `svaret` som `x` är. T.ex. kan ni räkna ut bokstavssumman i båda strängarna och returnera skillnaden. (Python-funktionen `ord` kan vara till nytta här.)

5. Modifiera `para` så att `barn` istället tar första halvan från `far` och andra från `mor` (eller något liknande, t.ex. var annat tecken från dem). OBS! Det är viktigt att alla strängar är lika långa, dvs. att `barn` blir lika lång som `mor` och `far`!

6. Modifiera `mutera` så att den väljer en slumpmässig bokstav i `barn` och ersätter den med ett slumpmässigt tecken från `alfabetet`

7. Kör programmet och se vad ni får som svar!

Problem 3. Du ska ordna bordsplaceringen till ett brölopp. Det är 100 deltagare som ska placeras runt 10 runda bord, varje med 10 platser. Du vill placera folk så att de har så mycket glädje av sina bordsgrannar som möjligt. Varje deltagare beskrivs av tre attribut: ålder (i år), kön (`M` eller `K`) och stjärntecken. Glädjen en person upplever beror på hur kompatibel hen är med sina två bordsgrannar och mäts i *glädjepoäng* (GP.)

1. Om grannarna har olika kön får de båda 10 GP. Om de har samma kön får de istället 3 GP.

2. Om skillnaden s mellan grannarnas åldrar är mindre än 15 så får de båda $15-s$ GP.

3. Om de har samma stjäntecken får de -5 GP. Har de närliggande stjärntecken (t.ex. vattuman och fisk) så får de 10 GP.

Skriv en genetisk algoritm som försöker maximera summan av allas GP.! En fil med gästernas informtion finns på `ethna.se/GästerGA.txt`

44. Förberäkningar

10 augusti

Ibland kommer man behöva beräkna liknande saker väldigt många gånger. Det kan då bli snabbare totalt sett att lägga en del tid på beräkningar innan man börjar svara på en ända fråga. Ett exempel som många av er säkert stött på är Erathostenes såll för att hitta alla primtal upp till ett visst tal, men det finns många tillfällen då förberäkningar gör att man snabbt kan svara på annars tidskrävande frågor.

≫ Kattis problem **howmanydigits**

Här ska vi svara på hur många siffror som ingår i $n!$ för 10 000 olika n upp till 1 000 000. Det kan underlätta att veta att antal siffror i ett tal x är `math.floor(math.log10(x))` och att $\log(x \cdot y) = \log(x) + \log(y)$

44.1 Prefixsummor

Givet en lista av tal $x_1, x_2, \ldots, x_n$ vill vi beräkna summan av talen från position L till men inte med R,

$$S(L,R) = \sum_{i=L}^{R+1} x_i.$$

Detta kan göras med $L - R$ operationer genom att beräkna summan, men om detta behöver göras många gånger är det inte tillräckligt snabbt. Om vi har förberäknat prefixsumman av listan, $P_i = \sum_{k=0}^{i}$, kan vi göra samma beräkning med en operation.

$$S(L,R) = P_R - P_L$$

Detta går också att utvidga till fler dimensioner.

44.2 Max över intervall i en lista

Om man vill hitta det högsta värdet mellan två positioner i en lista kan man leta igenom alla värden mellan dessa positioner. Om det är en lång lista kan det ta lång tid, så om man har många liknande frågor kan det bli för långsamt. Man kan då istället förberäkna ett segment-träd som jag visar på tavlan för att snabbt kunna hitta maxvärdet över vilket intervall man vill.

Om man bara är intresserad av intervall som börjar i ena kanten av listan så går detta att beräkna för alla slut på intervallet med ett ända svep. Använd detta för att lösa följande problem.

≫ Kattis problem **pivot**

44.3 Extrauppgifter

Se om ni kan lösa uppgifterna nedan

≫ Kattis problem **birdrescue**

≫ Kattis problem **reseto**

VI

Appendix

A. Regler Matematisk Yatzy

Matematisk Yatzy är en tävling där man tävlar i att lösa matematikproblem. Målet i tävlingen är att få så mycket poäng som möjligt (dels för lösta problem, dels för bonusar). Ni tävlar i lag om 3-4 personer.

Problemlösning. Varje lag får en uppsättning på 5 olika teman med 5 problem i varje. Problemsvaren redovisas i ordning inom temat, från 1:a till 5:e (till exempel godtas inte svaret till det 4:e problemet om inte svaren på 1:a, 2:a och 3:e problemet på samma tema har lämnats in). På varje problem har man ett försök att redovisa svaret. Om laget redovisar korrekt svar, så får man så mycket poäng som problemet är värt. Om svaret är felaktigt eller ofullständigt (t.ex. man anger några korrekta svar, men inte alla), så får man 0 poäng.

Första problemet i temat är alltid värt 10 poäng, andra är värt 20 poäng och så vidare till femte problemet som är värt 50 poäng. (Således kan man som mest få 750 poäng om man inte räknar med bonusar).

Bonusar. Varje lag kan också tjäna bonusar:

- Om alla problem inom ett tema hade korrekt svar, så får man ytterligare 50 poäng (”vågrätbonus”).
- Om alla problem med samma värde hade korrekt svar, så får man ytterligare det värdet som bonus (”lodrätbonus”).

Spelets slut. Spelet pågår i 90 minuter. Spelet tar slut för ett lag antingen om tiden är slut eller om man har försökt redovisa svaret på alla problem.

B. Svar Matematisk Yatzy Blå gruppen

Kombinatorik

1. (10) 21
2. (20) 4
3. (30) 24
4. (40) 10
5. (50) Simon, 54 fler

Textuppgifter

6. (10) 18
7. (20) 30
8. (30) 5
9. (40) 99
10. (50) 23

Algebra

11. (10) 3
12. (20) 3600
13. (30) -3
14. (40) (2,3),(-2,3),(3,2),(-3,2)
15. (50) 0,25

Heltal

16. (10) 121
17. (20) 4x4 och 3x6
18. (30) 3
19. (40) 4, 6, 12
20. (50) 2858

Geometri

21. (10) Kolla svaret
22. (20) $72^\circ, 72^\circ, 36^\circ$
23. (30) Arean ska vara 6 rutor.
24. (40) 108°
25. (50) 36, 45, 63, 72

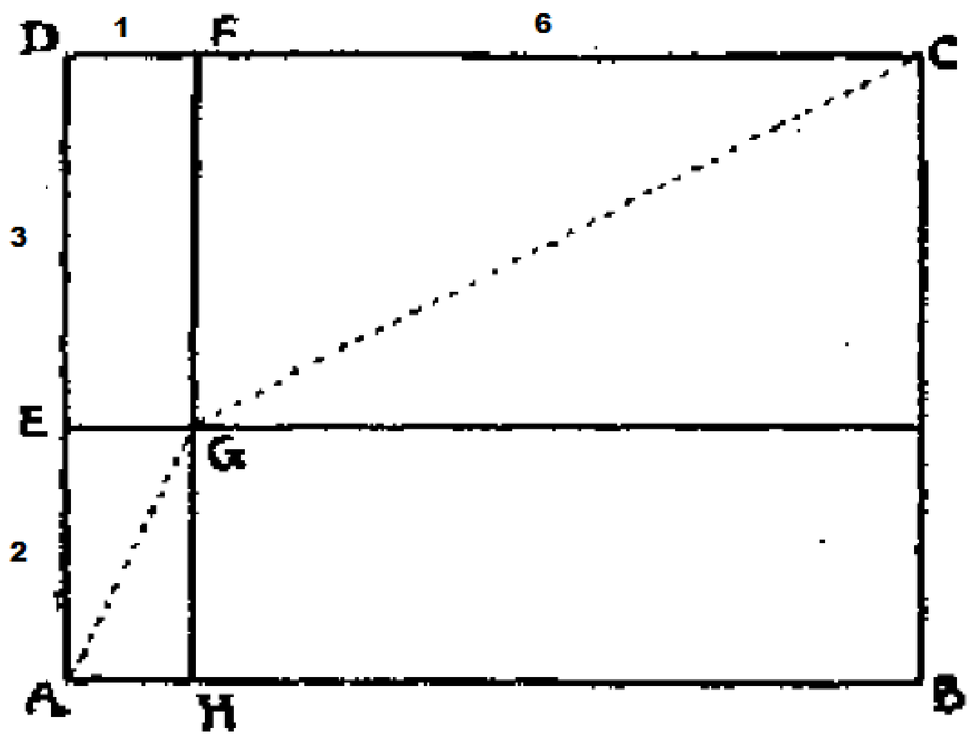
C. Svar Matematisk Yatzy Lila gruppen

Algebra

1. (10) (-1,1,-1,1...,-1,1)
2. (20) 0
3. (30) 2/27
4. (40) (0,0,0) och (1,1,1)
5. (50) 2

Geometri

6. (10)



7. (20) $(\sqrt{S_1}+\sqrt{S_2})^2$
8. (30) 2019

9. (40) $2\sqrt{3}-3$
10. (50) $7,2$

Talteori

11. (10) 195, 498, 265, 164
12. (20) 1, 153, 370, 371, 407
13. (30) 201
14. (40) 2857
15. (50) $p+q-1$

Kombinatorik

16. (10) $n=4$ och $n=6$
17. (20) $n=2^k-1$, där k är ett naturligt tal
18. (30) 2^{1010}
19. (40) 36
20. (50) n+2

Schackuppgifter

21. (10) $(2020*2020-2)/2=2040199$
22. (20) 2
23. (30) 36
24. (40) 47
25. (50) $28+36\sqrt{2}$

D. Regler Mattedrabbning

Regelsammanfattning Mattedrabbning

- Lagen turas (nästan alltid) om att utmana varandra på problem.
- Laget som blir utmanat presenterar lösningen eller skickar utmaningen tillbaka.
- En person skickas till tavlan för att presentera lösningen.
- Laget som inte presenterar lösningen skickar fram en opponent till tavlan, som ställer frågor.
- När opponenten är nöjd, så ställer juryn (lärarna) frågor.
- Man får poäng utefter hur rätt man hade när man presenterade.
- Om lösningen är helt korrekt, får det presenterande laget 12 poäng och andra får 0 poäng.
- Om lösningen är helt fel och opponenten påpekar det och förklara varför det är fel, så får laget som presenterar 0 poäng och laget som opponerar får 6 poäng.
- Man får inte prata med laget om man står framme vid tavlan.
- Man får ta halvminutspauser (max 6 stycken) för att laget ska få prata med personen som står framme vid tavlan.
- Ingen person får gå fram och presentera mer än två gånger.
- I början av drabbningen har man en liten s.k. ”kaptenstävling” där det vinnande laget får bestämma vem som utmanar först.
- Laget med mest poäng vinner!

Detaljerade regler Mattedrabbning

Mattedrabbning är en tävling för två lag i lösning av matematikproblem. Drabbningen består av två delar. Först får lagen ett antal problem och en bestämd tid att lösa dem på. Under lösningstiden får lagen inte använda hjälpmedel (förutom simpel miniräknare som är ok, samt pennor, passare, linjal och dyl.), och får endast prata om problemen med juryn. Efter utsatt tid börjar själva drabbningen, när lagen presenterar lösningarna för varandra enligt reglerna. När det ena laget presenterar en lösning, opponerar det andra laget på lösningen, d.v.s. söker fel och brister i den, och om det visar sig att lösningen saknas, får även det andra laget presentera sin egen lösning. Juryn tilldelar poäng både för presentation och opponering. Om lagen inte lyckas slutföra lösningen under diskussionen eller om de har några fel, får juryn ge en del av poängen (och även alla poäng) till sig själv. Om totalpoängen skiljer sig med mindre än 4 poäng när tävlingen är över, slutar drabbningen oavgjort, annars vinner laget med flest poäng.

Utmaning

Drabbningen består av flera omgångar. I varje omgång är det ena laget kärande och det andra laget svarande. Först utmanar käranden till ett problem, som inte presenterats tidigare (till exempel: "Vi utmanar motparten till problem 8"). Sen bestämmer det svarande laget om det tackar ja eller nej till utmaningen (de får 1 minut på sig att bestämma sig). Om laget tackar ja, måste det skicka en föredragande som ska presentera lösningen, medan det kärande laget får skicka en opponent med uppdraget att söka fel i lösningen. Om svarande laget tackar nej (det kallas för kontroll av korrekthet) måste det kärande laget skicka den föredragande, medan det svarande får skicka en opponent. (Ett lag får också låta bli att skicka en opponent för att spara ett uppträdande. Då deltar laget inte längre i den pågående omgången, får inte poäng och kan inte ångra sig).

Föredraget

Först presenterar den föredragande sin lösning. Han/hon måste ge svar på alla frågor i problemet och visa att alla svaren är riktiga och fullständiga. Bland annat måste den föredragande bevisa varje påstående den formulerar eller hänvisa till det som välkänt. Den föredragande måste uttrycka sig klart, bland annat måste han/hon vara beredd att upprepa vilken del som helst av sitt föredrag om juryn eller opponenten ber om det. Föredragets tid är begränsat till vanligtvis 10-15 minuter. Om tiden tar slut men den föredragande vill fortsätta får juryn bestämma om han/hon får göra det.

Den föredragande får ha med sig ett pappersark med bilder, och (om juryn tillåter det) ett med beräkningar. Men man får inte ha med sig texten med hela lösningen. Den föredragande får:

- före föredraget rita upp allt som behövs på tavlan (teckningar, ritningar o.s.v.);
- låta bli att svara på opponentens frågor före diskussionen;
- be opponenten att precisera sin fråga (till exempel kan man förelägga sin egen version ”Förstår jag rätt att du frågar ...?”);
- låta bli att svara på en fråga, med någon av motiveringarna: a) det vet jag inte; b) jag har ju redan svarat på denna fråga (och förklara när och hur); c) det är inte en korrekt fråga, eller den frågan ingår inte i vetenskaplig diskussion. Om opponenten inte godkänner motivering (b) eller (c) bestämmer juryn vem som har rätt.

  Den föredragande måste inte:

- förklara hur man har kommit på svaret om man kan bevisa på ett annat sätt att svaret är riktigt och fullständigt;
- jämföra sitt eget förfaringssätt med ett annat, bland annat i vilket mån det är kort, vackert eller generellt.

Opponering

Under föredraget får opponenten bara ställa frågor om den föredragande tillåter det. Dock får opponenten be om att upprepa en del av föredraget, och även tillåta att ett påstående som man tror vara självklart inte behöver bevisas. Efter föredraget får opponenten ställa frågor. Om opponenten inte kan ställa nästa fråga inom en minut bestäms det att man inte har fler frågor. Om den föredragande inte börjar svara på en fråga inom en minut bestäms det att svaret saknas. Som en fråga får opponenten:

- kräva att den föredragande upprepar en viss del av föredraget;
- be den föredragande att precisera vilket påstående som helst, bland annat genom att a) be om en terminologisk definition (”Vad menar du med...”) eller b) formulera om ett påstående och be om en bekräftelse (”Förstå jag rätt att du påstår...”);
- be den föredragande att bevisa ett påstående som den föredragande har använt och som opponenten inte känner till (juryn bestämmer om påståendet verkligen är allmänt känt);
- godkänna svaret eller icke godkänna det (i senare fall med motivering).

Om opponenten tror att den föredragande drar ut på tiden för att finna en lösning under föredraget, eller att föredraget mest inte har något att göra med lösningen, då får opponenten efter 10 minuter be den föredragande att framlägga ett svar (om problemet måste ha ett svar) eller en plan för sitt resonemang. Opponenten måste upprepa och precisera sina frågor om den föredragande eller juryn ber om det. Slutligen får opponenten ge ett utlåtande på en av följande former: a) lösningen är riktig; b) lösningen (svaret) är riktig på det hela taget, men det finns dessa brister; c) lösningen (svaret) är felaktig, det finns fel (brister) i dessa nyckelpåståenden, eller det finns ett motexempel.

Om opponenten godkänner lösningen, slutar han/hon och hans/hennes lag att delta i diskussionen under den här omgången. Vanligtvis börjar juryn ställa frågor till den föredragande efter opponentens utlåtande. Juryn får även blanda sig i diskussionen tidigare, dock får den då inte ställa frågor som hade kunnat ge poäng till opponenten om denne skulle ställa dem.

Båda lagen måste vara tysta under föredraget och diskussionen. För kommunikation och rådgivning mellan ett lag och dess representant får det tas en paus i en halv minut. (Motparten får använda denna tid för konsultation också.) Varje lag får ta en sådan paus när som helst, dock bara 6 gånger under drabbningen. Det är kaptenen som tar pausen. Representanten vid tavlan får bara be kaptenen att ta pausen eller be om att få bli utbytt mot en annan spelare (detta räknas som två pauser).

Antalet uppträdanden

Varje spelare har rätt att uppträda två gånger, d.v.s man får skickas till tavlan som fördragare eller opponent högst två gånger under drabbningen. Om laget byter ut sin representant vid tavlan räknas det som uppträdande av både den nya föredragande och den som blivit utbytt. Dessutom kostar ett utbyte två halvminutspauser (denna minut får laget använda även för konsultation). Således kan ett lag göra högst tre utbyten.

Den första utmaningen – Kaptenstävlingen

Vem som skall utmana först avgörs genom kaptenstävlingen just före själva drabbningen. Varje lag skickar en representant till tavlan (det kan vara kaptenen eller vilken lagmedlem som helst). Juryn ger ett enkelt problem till representanterna. Representanten som först säger sig ha ett svar, får svara. Man vinner om lösningen är rätt och förlorar annars. Juryn måste säga i förväg om det räcker med bara svar eller om det krävs ett svar med resonemang. Istället för kaptenstävlingen får juryn använda en lottdragning eller framställa ett spel för representanterna.

Rollomvändning

Rollomvändning kan bara ske i den omgång som inte är kontroll av korrekthet. Om juryn understödjer opponentens kritik, erbjuder juryn opponenten att rätta felen och fylla luckorna i föredraget. Om opponenten tackar ja sker en partiell rollomvändning och opponenten och den föredragande byter roller. Om juryn understödjer opponentens påstående att lösningen saknas i föredraget, eller är helt felaktig, erbjuder juryn opponenten att presentera sin egen lösning. Om opponenten tackar ja sker en total rollomvändning. Efter rollomvändningen får den före detta opponenten poäng för sin lösning, medan den före detta föredragande kan få poäng för kritik. Dock kan det inte ske rollomvändning en gång till i samma omgång.

Korrekt utmaning

Om utmaningen antas direkt, så är den alltid s.k. *korrekt*. Om utmaningen skickas tillbaka, så finns det två möjligheter. Om det utmanande laget erkänner att de inte har någon lösning, så är utmaningen automatiskt *inkorrekt*.

Om det utmanande lagets föredragare går fram till tavlan och berättar en lösning så beror korrekheten på. Om opponenten kan bevisa att lösningen är inkorrekt så räknas även utmaningen som inkorrekt. Om opponenten godkänner lösningen räknas utmaningen som korrekt.

Ordningen på utmaningarna och mattedrabbningens slut.

Om utmaningen som skedde var inkorrekt, så måste det utmanande laget utmana även i nästa omgång. I alla andra fall turas lagen om att utmana varandra.

När som helst kan det utmanande laget säga att de inte längre vill utmana denna mattedrabbning (vanligtvis gör man detta när man inte längre har några lösta problem bland de kvarvarande). Efter det får det andra laget möjligheten att redovisa de kvarvarande problemen (vilka de vill, de måste inte redovisa något). Laget som sagt att de inte längre vill utmana kan då bara opponera och få poäng för opponering. Inga rollbyten kan längre ske.

Mattedrabbningen är slut när alla problemen har redovisats eller om ett av lagen säger att de inte vill utmana längre och det andra laget vill inte längre redovisa något.

Poängutdelning

Varje problem kan ge 12 poäng, vilka slutligen delas mellan den föredragande, opponenten och juryn. Om den föredragande har presenterat en rätt och fullständig lösning, får han/hon alla 12 poäng. Annars finns det fel och luckor i lösningen, som opponenten måste upptäcka och påpeka (om opponenten bortser från några fel måste juryn påpeka dem senare).

Den föredragande får rätta felen och fylla i luckor. Orättade fel och ofyllda luckor minskar hans/hennes slutpoäng (juryn bestämmer med hur mycket). Dock om den föredragande har rättat alla fel och har presenterat den fullständiga lösningen bara efter ledande frågor från opponenten och/eller juryn, kallas det en ”oren lösning”. Då får juryn ta 1-2 poäng av den föredragande och ge dem till opponenten eller till sig själv. För varje orättad brist som opponenten har påpekat får han/hon hälften av bristens kostnad (d v s halva den poäng som avdragits från den föredragande). Den andra halvan kan opponenten försöka få genom att rätta bristerna själv (om juryn tillåter en partiell rollomvändning).

Om de påpekade och orättade bristerna kostar sammanlagt 6 poäng eller mer, bestäms det att opponenten bevisat att lösningen saknades. Då kan juryn tillåta en fullständig rollomvändning, d v s opponenten får presentera sin egen lösning helt från början. Efter rollomvändning tilldelar juryn de poäng som är kvar enligt samma princip: f.d. opponenten får för sitt föredrag, den f.d. föredragande för kritik (halvkostnadsprincipen och oren lösning gäller). Om opponenten har bevisat att lösningen saknas under kontroll av korrekthet får denne 6 poäng, medan den föredragande får 0 till 6 poäng för idéer.

Häftet innehåller material från Mattekollo 2019 för åk 9-gy2, alla matematik- och programmeringsgrupper. Den blåa gruppens tema var invarianter, den lila reell analys. Specialkursen om inversion är inkluderad.

Tack till alla deltagare och ledare!