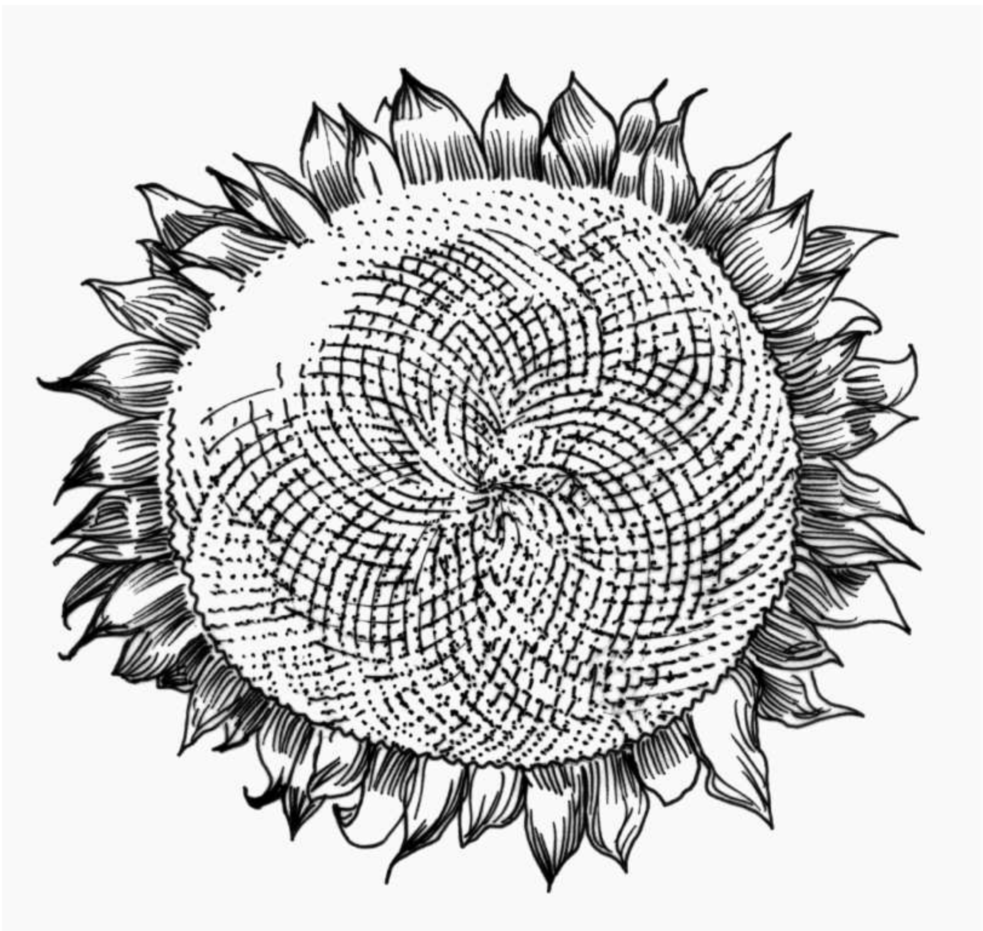
Solmu

Matematiikkalehti
1/2004

http://solmu.math.helsinki.fi/



Solmu 1/2004

ISSN 1458-8048 (Verkkolehti)
ISSN 1459-0395 (Painettu)

Matematiikan ja tilastotieteen laitos
PL 4 (Yliopistonkatu 5)
00014 Helsingin yliopisto

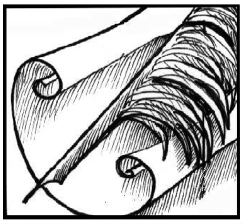
http://solmu.math.helsinki.fi/

Numeroon 2/2004 tarkoitetut kirjoitukset pyydämme lähettämään maaliskuun 2004 loppuun mennessä.

Kiitämme taloudellisesta tuesta Jenny ja Antti Wihurin rahastoa sekä Suomen Kulttuurirahastoa.

Huom! Solmun paperiversio postitetaan vain niihin kouluihin, jotka ovat sitä erikseen pyytäneet. Solmun Internet-sivuilta saatava paperiversio on mahdollista tulostaa omalla kirjoittimella. Toivomme, että lehti ei jää vain opettajien luettavaksi, vaan sitä kopioidaan kaikille halukkaille.

Sisällys



Matematiikan päivät ja matematiikan aseman parantaminen

Suomen matemaattinen yhdistys ja Oulun yliopiston matemaattisten tieteiden laitos järjestivät Oulussa tammikuun 2004 alussa Matematiikan päivät. Päivillä oli runsaasti tilaisuuksia, jotka sopivat matematiikasta ja sen opettamisesta kiinnostuneille. Matematiikan tutkimuksen sessiossa saatiin mm. kuulla, miten noin 160 vuotta vanha Catalanin probleema ratkesi vuonna 2002. Ongelma on helppo ymmärtää – vaan ei ratkaista – peruskoulutiedoilla.

Opetussessiossa esiteltiin monipuolisesti opetuksen kehittämishankkeita, mm. verkko-opetusta, tuutortoimintaa sekä matematiikkakerhoja ja -leirejä. Erittäin suosituiksi tulleista matematiikkakerhoista kertoivat nuoret oululaiset opettajat ja opiskelijat, Oulussa on päästy ilahduttavasti alkuun tässä tärkeässä toiminnassa. Toivottavasti kipinä leviää muuallekin Suomeen. Solmuun tullaan keräämään ideoita, kokemuksia ja vinkkejä siitä, millaista aineistoa on saatavana. Toivottavasti yhteistyö lähtee hyvin käyntiin ja yhteiseen kekoon kannetaan korsia.

Matemaattisista videoista saattoi aistia matematiikan kauneuden. Eräs niistä kertoi Granadassa sijaitsevan Alhambran seinäkoristeiden yhteydestä algebraan, siis tässä on eräs esimerkki algebran ja geometrian yhteydestä.

Matematiikka ja media -paneeli oli järjestetty matematiikkakuvan laajentamista ja matematiikkaa koskevien virheellisten käsitysten korjaamiseksi. Tilaisuus oli erittäin tarpeellinen ja onnistunut. Yhteistyöideoita kehiteltiin, mutta valitettavasti tilaisuus kiinnosti vain harvoja toimittajia. Paneelin ja siihen liittyvien keskustelujen pohjalta koottiin yhdeksän kohdan muistilista matematiikan nostamiseksi sille kuuluvaan arvoon.

OULUN YLIOPISTO

Pääkirjoitus

1) Kulttuurikäsite on saatava laajemmaksi; myös luonnontieteet ja matematiikka ovat yhteistä kulttuuriperintöämme. Suomen selviytymisstrategiana mainitaan aina korkean tegnologian innovaatiot – samalla pitäisi tiedostaa, että matematiikka on korkean teknologian perusta. Lisäksi se on tarpeellinen yhä useammilla, myös ns. pehmeillä aloilla, kuten sairaanhoidossa ja biologiassa.

2) Suomen kielen on säilytettävä myös tieteen kielenä. Tämä on yhteiskunnallinen tehtävä. Maailmankulttuurien käsitteistä on voitava puhua suomeksi – ei riitä, että asiantuntijat keskustelevat keskenään englanniksi.

3) Koululaisille on esiteltävä laajat uravalinnat myös mediassa. Luonnontieteen ja matematiikan osaajia tarvitaan nykyistä paljon enemmän, mutta myös esimerkiksi kädentaitojen hallitsijoita.

4) Tiedekirjat on saatava näkyviin mediassa; suomenkielellä ilmestyvistä tiedekirjoista – myös luonnontieteen aloilta – on saatava arviot esimerkiksi päivälehtiin. Kirjakaupoissa tulisi olla nähtävänä suomenkielisiä tiedekirjoja ja matematiikan harrastuskirjoja. Syyksi niiden laiminlyömiseen ei riitä, ettei niiden myynti ole suurta, kirjakaupoilta toivotaan myös sivistystehtävää.

5) Toimittajien ja asiantuntijoiden on pyrittävä tekemään yhteistyötä.

6) Matematiikan perusosaaminen on tarpeen kaikille, esimerkiksi prosenttilaskut, murtoluvut, vertailut ja tilastojen esittämisen perusasiat ovat näitä. Koulukursseissa on nykyisellään tarpeeksi tai jopa liikaa matematiikan sisältöjä. Lisäpanostusta tarvitaan näiden sisältöjen kunnolliseen omaksumiseen.

7) Suomenkielellä verkossa ilmestyvä matematiikkalehti Solmu on saatava laajojen piirien tietoon: http://solmu.math.helsinki.fi

8) Matematiikka on monilla aloilla tarpeellista perusosaamista, hyvin työllistävänä se ehkäisee syrjäytymistä. Matematiikka on tasa-arvokysymys, sillä sen opiskelu antaa myös naisille mahdollisuuden valita perinteisesti miehisiä aloja. Matematiikka on hyvä kasvattaja, sillä sen opiskelu vaatii kurinalaisuutta ja pitkäjänteisyyttä sekä harjoittaa keskittymiskykyä ja opettaa ajattelua.

9) Matematiikka on kiehtovaa ja hauskaa. Sen todistavat kaikki sen koukkuun jääneet. Se on ”tervettä huumetta”.

Alli Huovinen
Lehtori
Matemaattisten tieteiden laitos
Oulun yliopisto

Marjatta Näätänen
Dosentti
Matematiikan ja tilastotieteen laitos
Helsingin yliopisto

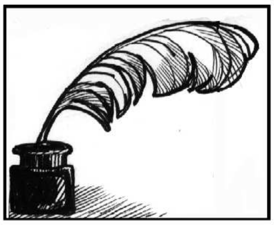
SMY:n matematiikkapalkinto Alli Huoviselle

Lehtori *Alli Huovinen* sai Matematiikan päivien avajaisissa julkistetun Suomen matemaattisen yhdistyksen matematiikkapalkinnon. Perustelut olivat seuraavat:

”Suomen matemaattinen yhdistys on päättänyt myöntää vuoden 2004 matematiikkapalkinnon lehtori Alli Huoviselle Oulun yliopiston matemaattisten tieteiden laitokselta. Matematiikkapalkinto on suuruudeltaan 3000 euroa.

Lehtori Huovinen on toiminut poikkeuksellisen aktiivisesti matemaattisen yleissivistyksen levittämiseksi ja tukemiseksi, erityisesti naisten ja nuorten piirissä. Oulun yliopistossa tekemänsä tärkeän ja innostavan kasvatustyön ja perusopetuksen kehittämisen lisäksi Huovinen on luonut, käynnistänyt ja ylläpitänyt yhteistyötä koulujen suuntaan, esimerkiksi matematiikkakerhotoiminnan muodossa. Kerhoja toimii tällä hetkellä n. 30, ja oppilaita niissä on n. 400, joista yli puolet tyttöjä. Kerhojen ja Huovisen pyyteettömän työn myötä innostus matematiikan alaa kohtaan on herännyt laajasti Oulun seudulla ja muualla Pohjois-Suomessa.”

Pääkirjoitus



Koulutuksen mallimaa?

Suomen menestyminen OECD:n tuoreimmassa koulutusvertailussa on huomioitu näyttävästi eturivin sanomalehtiä myöten. Vuoden 2004 helmikuun alussa julkaistu tutkimus käsittelee informaatio- ja viestintäteknologian (ICT) hyödyntämistä ja osaamista toisen asteen koulutuksessa, siis lukioissa ja ammattikouluissa. Suomen lisäksi mukana vertailussa olivat Belgia, Espanja, Irlanti, Italia, Korea, Meksiko, Norja, Portugali, Ranska, Ruotsi, Sveitsi, Tanska ja Unkari.

Verkkosivuillaan www.oecd.org OECD otsikoi tutkimuksen yhteenvedon "OECD havainnoi pettymyksen informaatio- ja viestintäteknologian käytössä toisen asteen koulutuksessa". *Helsingin Sanomien* (4.2.2004) otsikko samasta tutkimuksesta antaa varsin toisenlaisen kuvan: "Lukioiden ja ammattikoulujen opinto-ohjaukselle kehut OECD:lta". 'Hyvä me' -henki korostuu uutisartikkelin alkavassa lauseessa "Suomi on taas koulutuksen mallimaita OECD-vertailussa". Innostusta osaamisen tasosta lievennetään kuitenkin jo parin päivän päästä peräti pääkirjoituksessa otsikolla "Koulujen hyvä sijoitus ei paljasta koko totuutta" (*HS* 6.2.2004).

Tutkimuksessa nostetaan esille erityisesti opinto-ohjauksen, tietokonelaitteistojen ja opettajien täydennyskoulutuksen hyvä taso Suomessa. Vähemmälle huomiolle – ainakin uutisoinnissa – jää ICT:n hyödyntäminen opetuksessa suhteessa niihin mahdollisuuksiin, jotka Suomessa on mittavalla panostuksella saavutettu. Totuus lienee, että kouluissa tietokoneita ei käytetä juuri muuhun kuin tekstinkäsittelyyn ja tiedonhakuun verkosta, nettisurffaukseen. Mitenkään näiden taitojen merkitystä väheksymättä ICT:n opetuskäytössä on vielä rutkasti kehittämistä niin matematiikan kuin muidenkin aineiden opettajilla. Tosin ICT:n mielekäs opetuskäyttö on erittäin vaativaa ja aikaavievää. Opettajat Suomessa ovat varmasti tehneet voitavansa, lyhyessä ajassa on tapahtunut valtava muutos, eikä ajan tasalla pysyminen ole ollut kenellekään helppoa.

* * *

Verkko-Solmussa artikkelit julkaistaan nykyisin vain PDF- ja PS-muodoissa, eikä uusista artikkeleista ole HTML-versioita enää saatavilla. PDF on muodostunut ladontaa vaativien verkkojulkaisujen standardiksi, ja toisin kuin HTML, se soveltuu hyvin myös matemaattisen tekstin esittämiseen verkossa. PDF-dokumenttien lukemiseen ja tulostamiseen tarkoitettu Adobe Reader -ohjelma on vapaasti kopioitavissa verkkosivulta www.adobe.com/products/acrobat/readstep2.html.

HTML-dokumenteista luopumista kokeiltiin Solmussa jo muutama vuosi sitten, mutta tuolloin verkkoyhteyksien hitaus ja PDF-tiedostojen toiminnan epävarmuus aiheuttivat paluun HTML-versioihin. Nyt näistä ongelmista on päästy eroon; jos olette käyttökokemustenne perusteella eri mieltä, niin lähettäkää palautetta.

Mika Koskenoja

Toimitussihteerin palsta

Matematiikka – monipuolinen tiede

Matti Vuorinen
Professori
Matematiikan laitos, Turun yliopisto

Mitä matematiikka on?

Lähes jokaisella suomalaisella on käsitys siitä, mitä matematiikka on kouluaineena. Matematiikan todellinen luonne samoin kuin modernin matematiikan tutkimuksen motivaatiot ja visiot jäävät kuitenkin useimmille epäselviksi tai kokonaan tuntemattomiksi. Nykyyhteiskunnan arkipäivässä matemaattisiin keksintöihin pohjautuvilla laitteilla on kuitenkin suuri merkitys, olkoonkin, että matematiikan osuus jää usein piiloon laitteen käyttäjältä. Mitä sitten tarkoitetaan sanalla ”matematiikka”? Turun Akatemiassa vuonna 1645 [5] eräässä prof. *Simon Kexleruksen* johdolla julkaistussa väitöskirjassa annettiin seuraava määritelmä [6, s. 47]:

”Matematiikka on taito tutkia ja selvittää demonstraatioiden avulla olioiden kvaliteetteja.”

En nyt pyri Kexleruksen tavoin määrittelemään matematiikkaa, vaan nostamaan esille joitakin modernin matematiikan tutkimuksen ajankohtaisia visioita ja tutkimusaloja.

Matematiikka tieteenä

Varhaisimmat dokumentit matematiikan historiasta on löydetty muinaisen Babylonian nuolenpääkirjoituksella tehdyistä savitauluista ja muinaisen Egyptin hieroglyfikirjoituksista. Kummassakin tapauksessa oli kysymys kokemusperäisten seikkojen soveltamisesta. Ratkaiseva askel eteenpäin otettiin antiikin Kreikassa, jossa kokemusperäiset geometriset seikat systematisoitiin ja niille esitettiin loogisen päättelyn avulla perustelu, todistus. Näin syntyi matemaattinen tiede [1, s. 768]. Teoksissaan *Eukleides* kokosi geometrian alalla saavutetut tulokset oppijärjestelmäksi, jossa keskeisiä olivat seuraavat seikat:

- tarkastelun kohteena olevat suureet määritellään täsmällisesti,
- tutkimustulokset lausutaan tarkassa muodossa,
- tulokset todistetaan deduktiivisesti, aksiomeihin palautuvilla loogisilla perusteluilla,
- tarkastelut eivät ole irrallisia, vaan liittyvät, uutta lisävalaistusta tuoden, aikaisempaan tutkimukseen.

Nämä piirteet ovat edelleenkin tyypillisiä kaikelle matemaattiselle tutkimustyölle. Matematiikan kehittymiselle tieteenä on olennaista, että uudet tulokset voidaan rakentaa vanhojen tulosten perustukselle, niitä hylkäämättä. Tehtävänasettelut voivat säilyttää ajankohtaisuutensa aikakaudesta toiseen samalla kuin ymmärrys niiden merkityksestä syventyy. Matematiikassa eivät ole keskeisiä numeroihin kohdistuvat laskutoimitukset tai kaavojen manipulointi, vaan matemaattinen päättely, jonka kohteina voivat olla esim. matemaattisiin struktuureihin, kuten yhtälöryhmien ratkeavuuteen, liittyvät tarkastelut. Matematiikan erikoistuneita tutkimusaloja on nykyään useita satoja algebrasta ja analyysistä tietotekniikan matemaattiseen teoriaan ja matemaattiseen fysiikkaan.

Mitkä seikat sitten ohjaavat matematiikan kehitystä? Virikkeinä voivat olla sovelluksissa, esim. fysiikassa, esiintyvät tarpeet. Monesti voidaan todeta, että tutkimustyössä karaistunut tutkijan sisäinen näkemys, intuitio, johtaa tutkimushypoteeseihin, joihin aikanaan toivottavasti löydetään myös ratkaisu.

Intuition merkitys matemaattisten ideoiden tienviittana ilmenee seuraavasta *Albert Einsteinista* kerrotusta esimerkistä [6]. Matemaattisen fysiikan alalla työskennellyt Einstein joutui ponnistelemaan vuosikymmenen ennenkuin lopulta keksi yleisen suhteellisuusteorian. Useista epäonnistuneista yrityksistään huolimatta, työn vielä ollessa kesken, hän oli kuitenkin vakuuttunut, että luonnon todellisuutta oikealla tavalla kuvaava ratkaisu oli olemassa ja se oli löydettävissä. Uskonsa oikean teorian saavutettavuuteen Einstein muotoili seuraavasti [6, s. 7]:

”Luonto näyttää meille vain leijonan hännän. En kuitenkaan epäile, että siihen liittyy leijona, vaikka hän ei heti pystykään paljastamaan itseään valtavan kokonsa takia.”

Ratkaisuihin pääsy voi joskus kestää pitkäänkin: mm. kysymystä paralleelipostulaatin asemasta geometriassa pohdittiin antiikin ajoista 1800-luvun alkuun, jolloin se ratkesi epäeuklidisen geometrian synnyn yhteydessä. Toisena esimerkkinä matemaatikkojen ammattikunnan sitkeydestä voi mainita, että kuuluisan 1600-luvulla esitetyn Fermat'n ongelman ratkaisuun päästiin vasta vuosikymmen sitten, yli 300-vuotisten ponnistelujen jälkeen.

Matematiikka ja sovellukset

Galileo Galilei toteaa tunnetussa lausumassaan 1500-luvulta, että luonnon kirjaa ei voi lukea tuntematta kieltä, jolla se on kirjoitettu [2]. Tämän kielen kirjaimet ovat kolmioita, ympyröitä ja muita geometrisia kuvioita ja ilman näiden tuntemusta on mahdotonta ymmärtää luonnon kieltä.

Näin siis Galilein aikana, mutta mitkä ovat matematiikan sovellusmahdollisuudet nykyään? Haluaisin alleviivata sanaa ”kieli” Galilein lausumassa. Voidaan nimittäin todeta, että matematiikan ilmaisuvoimainen ja täsmällinen käsitekieli tarjoaa sopivia työvälineitä käsitteenmuodostukseen, metodinkehitykseen ja tieteellisiin läpimurtoihin kokonaan uusilla tieteenaloilla.

Ensimmäisenä esimerkkinä otan esille *Bernhard Riemannin* 1800-luvun puolivälissä luoman geometrian keskeisen roolin Einsteinin suhteellisuusteoriassa kaksi sukupolvea myöhemmin. Geometrian ja matemaattisen fysiikan hedelmällinen vuorovaikutus jatkuu fysiikan uusimmissa teorioissa nykyäänkin.

Toisena esimerkkinä voisi mainita algoritmien matemaattisen teorian, joka sijoittuu matematiikan ja tietojenkäsittelytieteen välimaastoon. Matemaatikkojen *John von Neumann* ja *Alan Turing* uraauurtavilla tietokoneiden toimintaperiaatetta koskevilla töillä 1900-luvun puolivälissä oli keskeinen vaikutus ensimmäisten toimivien tietokoneiden syntyyn. Sittemmin algoritmien matemaattinen teoria on eriytynyt omaksi tutkimusalueekseen, jonka kehitys on ollut nopeaa viimeisten kolmen vuosikymmenen aikana. Yksittäisenä esimerkkinä alan suuresta merkityksestä voidaan mainita kaikkien Internetin käyttäjien tuntema hakupalvelu Google, jonka tehokkuus perustuu valtavan suuren matriisin ominaisarvojen laskentaan.

Kolmas esimerkkini on uusi tieteenala, tieteellinen laskenta. Kysymyksessä on noin kymmenen vuotta sitten omaksi alakseen eriytynyt matematiikan haara, jolla on vahva poikkitieteellinen luonne. Siinä yhdistyvät matematiikka ja tietojenkäsittelytiede. Sen piiriin kuuluvat mm. simulointi, matemaattinen mallinnus ja laajojen aineistojen tietokoneavusteinen data-analyysi. Simuloinnissa tutkimuksen kohteena olevaa ilmiötä kuvataan sen olennaiset piirteet sisältävän mallin avulla. Simulointitekniikoilla on erittäin laajat sovellusmahdollisuudet teknisiin tieteisiin, biotieteisiin ja luonnontieteisiin. Simulointimallit antavat apuvälineen esim. matkaviestinverkkojen komponenttien optimoinnille jo suunnitteluvaiheessa, jolloin kalliilta korjauskustannuksilta verkon rakennusvaiheen aikana vältytään. Kustannussäästöt ovat oleellisia kansainvälisen kilpailukyvyn säilyttämiseksi. Tieteellisen laskennan kehittyessä matematiikan tutkimukseen soveltuvat ohjelmistot ovat tulleet laajaan käyttöön. Nämä ohjelmistot ovat avaneet kokonaan uudenlaisia mahdollisuuksia kokeellisen aineiston käyttöön matematiikan tutkimuksessa ja opetuksessa.

Matematiikan haasteet

Ihmisen elinympäristö on muuttunut radikaalisti viime vuosikymmeninä. Osaltaan tämä johtuu arkipäivää helpottavista tekniikan keksinnöistä, joista edellä jo mainittiin tietokoneet ja matkaviestimet. Kummassakin tapauksessa matematiikan rooli on ollut keskeinen, seikka, jonka soisi olevan laajemmin tunnettu. Kysymys ei ole yksittäistapauksista, vaan arvostettujen asiantuntijatahojen näkemyksen mukaan high tech on suuressa määrin matemaattista tekniikkaa. Galileita mukaillen voisi todeta, että matematiikka on nykyajan ympäristöoppia.

Nokian tutkimusjohtaja *Yrjö Neuvo* piti syksyllä 2001 esitelmän otsikkona ”Mathematics for mobile generations” [9], jossa hän toi esille matematiikan tärkeän roolin nykyisten matkaviestimien kehityksessä ja totesi, että seuraavan sukupolven viestimissä matematiikan rooli edelleen kasvaa. Matkaviestimien tulevien sukupolvien kehitykseen osallistuminen on haaste paitsi matematiikalle yleensä, myös nimenomaan suomalaiselle matematiikalle, koska kyseessä on kansantaloudellemme merkittävä teollisuudenala. On paikallaan korostaa, että luotettavassa tiedonsiirrossa keskeisen tärkeät matematiikan alat, kryptografia ja koodausteoria ovat juuri Turun yliopiston matematiikan laitoksen vahvuusaloja. Akateemikko *Arto Salomaan* ja professori *Aimo Tietäväisen* perustamat tutkimusryhmät ovat luoneet uraauurtavaa tutkimusta maassamme ja heidän ryhmistään polveutuvat lähes kaikki näiden alojen suomalaiset asiantuntijat.

Paavo Lipposen pääministerikaudellaan asettama Valtion tiede- ja teknologianeuvosto toteaa uusimman v. 2003 julkaistun raporttinsa [10] yhteenveto-osassa matematiikan merkityksestä mm. seuraavaa: ”Suomen kansainväliseksi vahvuudeksi katsotun korkean koulutustason turvaamiseksi esitetään tietoyhteiskunnan perusvalmiuksiin panostamista, matematiikan ja luonnontieteiden osaamisen lisäämistä sekä tohtorintutkintoa seuraavaan tutkijanuraan ja tutkijoiden urakehitysmahdollisuuksiin panostamista.” Näin tämä tiedepolitiikan tuleviin linjauksiinkin todennäköisesti vaikuttava asiakirja tiivistää matematiikan merkityksen yhtenä tietoyhteiskunnan kulmakivistä.

Matematiikan asiantuntijoita on Suomessa liian vähän. Suurten ikäluokkien jäädessä eläkkeelle seuraavan vuosikymmenen aikana matematiikan, samoin kuin monien muiden alojen, asiantuntijatarpeen voi ennakoida merkittävästi vielä kasvavan sekä oppilaitoksissa että teollisuuden toimialalla. Keskeisinä tehtävinä ja tavoitteina tulevat olemaan toisaalta määrätietoisena jatkuva panostaminen tutkijankoulutukseen ja toisaalta yhteistyön syventäminen teollisuuden kanssa.

Lopuksi

Matematiikka on toisaalta menetelmätiede, jolla on yhdenmukaistava vaikutus useilla käyttöalueillaan, toisaalta yhtenä vanhimmista tieteistä se on enemmän kuin sovellustensa summa. Matematiikka on dynaaminen tiede, josta ehtymättä kumpuaa uusia ideoita sekä perustutkimuksen että sovellusten käyttöön. Kukin tiede tarjoaa ikäänkuin oman ikkunansa totuuden katsomiseen, ponnistelee kohti meitä ympäröivän todellisuuden ymmärtämistä ja antaa sovellustensa kautta motivaatioita ja visioita tulevaisuuden kehittämiseen. Matematiikan tarjoamat visiot liittyvät vahvasti luonnontieteiden ja tekniikan menetelmiin, joiden kehitykseen se on keskeisesti vaikuttanut antiikista nykypäivään.

Kirjallisuutta

[1] H. Bass: The Carnegie Initiative on the Doctorate: The Case of Mathematics, Notices of Amer. Math. Soc., Vol. 50, Nr. 7, Aug. 2003, 767–776.

[2] E.T. Bell: Matematiikan miehiä, WSOY, 1963.

[3] A. Borel: On the place of mathematics in culture. Duration and change, 139–158, Springer, Berlin, 1994.

[4] H. Hasse: Mathematik als Wissenschaft, Kunst und Macht, Wiesbaden, 1952.

[5] M. Klinge *et al.*: Helsingin yliopisto 1640–1990. 1. osa: Kuninkaallinen Turun akatemia 1640–1808. Otava, Helsinki, 1987, ISBN 951-1-09736-9.

[6] R. Lehti: Leijonan häntä − luoko tietoa luonto vai ihminen? Ursa, 2001, ISBN 952-5329-10-0.

[7] O. Lehto: Matemaattiset tieteet. Teoksessa: Suomen tieteen historia. osa. Luonnontieteet, lääketieteet ja tekniset tieteet. Päätoim. P. Tommila, Porvoo, 2000, ISBN 951-0-23108-8.

[8] Mathematics unlimited − 2001 and beyond, Ed. B. Engquist ja W. Schmid, Springer, Berlin, 2001, ISBN 3-540-66913-2.

[9] Y. Neuvo: Mathematics for mobile generations. Fifth Diderot Mathematical Forum, November 22, 2001. http://www.math.hut.helsinki.fi/diderot2001/programme.html

[10] Valtion Tiede- ja teknologianeuvosto: Osaaminen, innovaatiot ja kansainvälistyminen. http://www.minedu.fi/tiede_ja_teknologianeuvosto/julkaisut/katsaus2003.pdf

Tämä artikkeli on Matti Vuorisen virkaanastujaisesitelmä Turun Akatemiatalolla 15.10.2003. Artikkeli on julkaistu *Arkhimedes*-lehden numerossa 6/2003, ja se julkaistaan Solmussa *Arkhimedes*-lehden luvalla.

Laskutaito ja numeroiden lukutaito edelleen tarpeen

Matti Seppälä
Luonnonmaantieteen professori
Maantieteen laitos, Helsingin yliopisto

Timo Tossavainen ja *Tuomas Sorvali* kirjoituksessaan "Matematiikka, koulumatematiikka ja didaktinen matematiikka" (*Tieteessä tapahtuu* 8/2003) kirjoittavat, että "Laskimien ja tietokoneiden kehitys on siis tehnyt puhtaasti teknistä laatua olevan laskutaidon käytännön kannalta tarpeettomaksi." Tästä sallittaneen muutama maallikon reunahuomautus, koska olen hieman eri mieltä. Mekaaninen laskutaito on monesti ainakin hyödyllistä. Väitteeni tueksi kerron muutaman esimerkin.

Jokunen vuosi sitten seisoin jonossa Snellmaninkadun nyttemmin suljetussa postissa Helsingin keskustassa ostaakseni muutaman arkin joulumerkkejä. Laskin aikani kuluksi 20 merkin arkin hinnan. Kun vuoroni lopulta tuli ja sain merkit, niin virkailija (silloin oli vielä postivirkailijoita) ilmoitti laskelmastani poikkeavan korkeamman hinnan. Minä kinaamaan vastaan, jolloin hän näytti arkin reunaan painetun hinnan (joka riville on laskettu kumulatiivinen hinta) ja viivakoodilla otettuna hinta oli ylimmälle riville painettu, mutta kun se oli väärä. Virkailija suostui lopulta jonosta huolimatta laskemaan koneella kahdenkymmenen merkin hinnan ja päätyi minun ilmoittamaani. Viimeinen rivi oli hinnaltaan kaksinkertainen. Lopulta sain maksetuksi oikean hinnan ja vitsailin, ettei kenenkään kannata ostaa viimeistä riviä merkkejä. Kuinka monet tuhannet ihmiset maksoivat sinä vuonna liikaa joulumerkeistään? Havaintoni jälkeen *Helsingin Sanomat* julkaisi nopeasti asiasta kirjoituksen ja posti muistaakseni pahoitteli asiaa, mutta varmaan ympäri maata painettua hintaa perittiin asiakkailta kauden loppuun.

Olen usein oikaissut kaupan kassalla yhteenlaskusummia, jotka voivat olla aivan mielikuvituksellisia. Koneet laskevat lukuja, joita niihin syötetään. Päässälaskutaidosta on hyötyä.

Arkikieleen on pesiytynyt epätäsmällisyyksiä, jotka leviävät taudin lailla, koska ihmiset eivät ajattele tai ymmärrä mitä he puhuvat. Sanotaan, että jokin asia on puolet suurempi kuin toinen ja tarkoitetaan, että se on kaksi kertaa niin suuri kuin toinen. Joku muu asia on muka kaksi kertaa pienempi kuin toinen, vaikka tarkoitetaan, että se on vain puolet toisesta. Jos kerrotaan kahdella, niin tapahtuu suurenemista, ja jos puolella kerrotaan, niin koko pienenee. Ainakin näin minulle on opetettu. Tässä olisi matematiikan opettajille työsarkaa. Jo TV:n uutistoimittajille pitäisi opettaa tätä joka päivän matematiikkaa.

Merkittävä parannus olisi myös, jos opittaisiin lukujen suuruussuhteiden ymmärtäminen, ettei puhuttaisi julkisuudessa aivan mahdottomia. Pinta-alojen ymmärtäminen on näköjään vaikeaa.

WWF:n pääsihteeri kertoi kerran radiossa miten sademetsiä hakataan 100 miljoonaa neliökilometriä vuodessa. Soitin hänellä ohjelman jälkeen ja kysyin mikä on Euroopan pinta-ala. Hänellä ei ollut aavistustakaan. Kun kerroin, että se on noin 10 miljoonaa neliökilometriä, hän myönsi ehkä erehtyneensä.

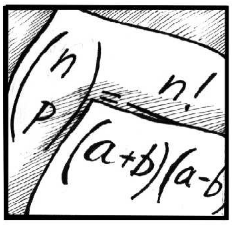
Ilmastonmuutosta päivittelevä professori kirjoitti *Helsingin Sanomissa* (1.9.2002) vieraskynässä miten ”Pohjois-Amerikassa lumipeitteen alta vapautuvia uusia alueita lasketaan muodostuvan vuosittain noin 103 miljoonaa neliökilometriä”. Kuitenkin koko Pohjois-Amerikan pinta-ala on vain 23,5 miljoonaa neliökilometriä.

Kerran maantieteen seminaarissa piti esitelmän opiskelija, joka ei osannut lukea 100 000:ta suurempia lukuja, joita hän esitykseensä oli kirjoittanut.

Käytännön laskutaito tuli vastaan myös niille opiskelijoille, jotka eivät millään tahtoneet saada selville peruskarttalehden pinta-alaa, kun karttaneliön sivut ovat 10 kilometriä.

Mielestäni matematiikan opettajilla on paljon hyvin paljon perustavanlaatuista opetettavaa ilman hienoja teorioita. Matematiikka on kieli ja kieltä ei voi käyttää ellei ole sanoja ja yhteisiä käsitteitä, joten kyllä ne on syytä oppia. Onko muuten olemassa matemaattista huumoria? Voisiko sitä käyttää opetuksessa?

Tämä artikkeli on julkaistu *Tieteessä tapahtuu* -lehden numerossa 1/2004, ja se julkaistaan Solmussa *Tieteessä tapahtuu* -lehden ja artikkelin kirjoittajan luvalla.



Solmun tehtäviä

Lähetä ratkaisusi Solmun tämänkertaisiin tehtäviin Solmun toimitukseen viimeistään kesän 2004 aikana joko sähköpostitse osoitteeseen

toimitus@solmu.math.helsinki.fi

tai kirjeenä osoitteeseen

Solmun toimitus
Matematiikan laitos
PL 4
00014 Helsingin yliopisto.

Parhaat ratkaisuehdotukset julkaistaan Solmun tulevissa numeroissa.

1. Kuusinumeroisesta luvusta vähennetään luvun numeroiden summa ja toistetaan sama operaatio saadulle tulokselle. Onko mahdollista, että lopputuloksena saadaan luku 2002?

2. Ratkaise yhtälö

$$|x+3| + p|x-2| = 5,$$

missä p on reaalinen parametri.

3. Nelikulmiossa $ABCD$ on

$AB = 1,\ BC = 2,\ CD = \sqrt{3},$ kulma $ABC = 120°$ ja kulma $BCD = 90°$.

Määritä sivun AD pituuden tarkka arvo.

4. Kolmion ABC sivun AB pituus on 10 cm, sivun AC pituus on 5,1 cm ja kulma $CAB = 58°$. Määritä kulman BCA suuruus asteen sadasosan tarkkuudella.

5. Millä todennäköisyydellä lottoarvonnassa (yksi arvonta viikossa) ainakin yksi seitsemästä tämän viikon lottonumerosta (numerot 1–39) arvottiin myös viime viikolla?

6. Joulupukki tarkkaili taivasta huolestuneena, syvästi mietiskellen. Seuraavana päivänä hän halusi matkustaa niin kauas kuin mahdollista jakaakseen lahjoja lapsille. Lopulta keskiyöllä alkoi sataa lunta. Lumisateen asiantuntijana hän näki heti, että sade oli sen laatuista, ettei se lakkaisi seuraavaan 24 tuntiin. Joulupukki myös tiesi, että ensimmäisen 16 tunnin aikana reki kulkee yhä nopeammin ja nopeammin (nopeus kiihtyy tasaisesti). Reki olisi alussa paikallaan, mutta kun 16. tunti olisi kulunut, lentäisi se kuin nuoli. Sen jälkeen kuitenkin matkan taivaltaminen vaikeutuisi yhä paksunevan lumen vuoksi, ja seuraavan 8 tunnin aikana reen huippunopeus laskisi tasaisesti takaisin nollaan. Toisaalta joulupukki ei kuitenkaan haluaisi uuvuttaa porojaan pakottamalla niitä yli 8 tunnin työhön. Milloin joulupukin pitäisi lähteä matkaan kulkeakseen mahdollisimman pitkän matkan?

7. Onko olemassa sellainen aritmeettinen lukujono, joka koostuu erisuurista positiivisista kokonaisluvuista, ja jossa mikään jonon termi ei ole jaollinen millään neliöluvulla, joka on suurempi kuin 1?

8. Onko jollakin neliöluvulla desimaaliesitys, jonka luvun numeroiden summa on 2002?

9. Ratkaise seuraava yhtälö:

$$2x^4 + 2y^4 - 4x^3y + 6x^2y^2 - 4xy^3 + 7y^2 + 7z^2 - 14yz - 70y + 70z + 175 = 0.$$

10. Ympyrä k_1, jonka säde on R, ja ympyrä k_2, jonka säde on $2R$, koskettavat ulkoisesti pisteessä E_3, ja ympyrät k_1 ja k_2 koskettavat ulkoapäin myös ympyrää k_3, jonka säde on $3R$. Ympyrät k_2 ja k_3 koskettavat pisteessä E_1, ja ympyrät k_3 ja k_1 koskettavat pisteessä E_2. Todista, että kolmion $E_1E_2E_3$ ympäri piirretty ympyrä on yhdenmukainen ympyrän k_1 kanssa.

11. Onko totta, että jos on olemassa annetun puolisuunnikkaan kantojen kanssa yhdensuuntainen suora, joka puolittaa sekä puolisuunnikkaan pinta-alan että ympärysmitan, niin silloin puolisuunnikas on suunnikas?

12. Tasakylkisten kolmioiden kärkikulmat ovat 140° ja ne on piirretty annetun kolmion ABC sivuille AB ja AC (ulkopuolelle). Uudet kulmapisteet ovat silloin A_1 ja C_1. Tasakylkinen kolmio, jonka kärkikulma on 80° pisteessä B_1, piirretään sivulle AC (ulkopuolelle). Määritä kulma $C_1B_1A_1$.

13. Suora katkaistu ympyräkartio on piirretty pallon ympärille. Mikä on sen tilavuuden ja pinta-alan suurin mahdollinen suhde?

14. Määritä kolmiulotteisessa koordinaatistossa kuutio, jonka särmät eivät ole yhdensuuntaiset koordinattiakselien kanssa, mutta niiden pituus on kokonaisluvun mittainen.

15. Anna ja Sofia heittävät vuorotellen arpakuutiota. Arpakuution osoittama luku lisätään aina kummankin erikseen keräämään pistemäärään. Pelin voittaa luvulla 4 jaollisen pistemäärän ensin saavuttanut pelaaja. Jos Anna aloittaa pelin, millä todennäköisyydellä hän tulee voittamaan sen?

16. Todista, että jos n on mielivaltainen positiivinen kokonaisluku, niin

$$\sum_{\substack{k_1,\ldots,k_n \geq 0 \\ k_1+2k_2+\ldots+nk_n=n}} \frac{(k_1+k_2+\ldots+k_n)!}{k_1!\cdot\ldots\cdot k_n!} = 2^{n-1}.$$

17. Kolmion ABC korkeusjanojen leikkauspiste on M, ja sen sisään piirretty ympyrä, keskipisteenä O, koskettaa sivuja AC ja BC pisteissä P ja Q. Todista, että jos M sijaitsee suoralla PQ, niin suora MO kulkee sivun AB läpi sen keskikohdasta.

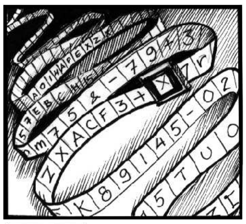
18. Olkoon a_n termin x^n kerroin polynomissa

$$(x^2 + x + 1)^n.$$

Todista, että jos $p > 3$ on alkuluku, niin

$$a_p \equiv 1 \pmod{p^2}.$$

Lähde: *KöMaL*, December 2002, *http://www.komal.hu/*.
Käännös ja ladonta: ***Jouni Koponen***



Ohjelmoinnin alkeita Python-kielellä

Antti Rasila
Tutkija
Matematiikan ja tilastotieteen laitos, Helsingin yliopisto

Johdanto

Tämän artikkelin tarkoituksena on esitellä lukijalle Python-ohjelmointikieltä matemaattisten sovellusten näkökulmasta. Artikkeli on tarkoitettu ohjelmoinnin peruskurssiksi, jossa käydään läpi yksinkertaisten ohjelmien kirjoittamista Python-kielellä, mutta sen voi vaihtoehtoisesti ymmärtää Pythonin yleisesittelyksi jo johonkin toiseen ohjelmointikieleen perehtyneelle lukijalle. Tarkoituksena on kirjoittaa artikkeliin myöhemmin jatkoa, jonka sisällöksi olen kaavaillut yksinkertaisten numeerisen matematiikan sovellusten kirjoittamista *Numerical Python* -kirjastoa käyttämällä sekä graafisten käyttöliittymien kirjoittamista ja matemaattisen datan visualisointia *Tkinter*-laajennuksen avulla.

Python on interaktiivinen olio-ohjelmointikieli, jossa yksinkertainen ja selkeä syntaksi yhdistyy kokeneemmankin ohjelmoijan arvostamiin monipuolisiin ominaisuuksiin. Python-kieli on helppo oppia, ohjelmien kirjoittaminen on sillä nopeaa ja virheiden etsiminen valmiista ohjelmasta vaivatonta. Python sopii hyvin ensimmäiseksi ohjelmointikieleksi ja on osoittautunut suosituksi monenlaisissa ohjelmointiprojekteissa.

Tavallisimmin Pythonia käytetään erilaisten tietoliikenne- ja ylläpitosovellusten kirjoittamisessa. Pythoniin on saatavissa laajennuksia, joiden avulla sitä voi tehokkaasti käyttää moniin muihinkin tarkoituksiin. Yksi tällainen laajennus on tämän artikkelin seuraavassa osassa esiteltävä *Numerical Python* -kirjasto, jonka avulla Python on laajennettavissa ominaisuuksiltaan lähes kaupallista MATLAB-ohjelmistoa vastaavaksi numeerisen matematiikan ohjelmointiympäristöksi. Osoitteesta `http://www.python.org` ladattavissa oleva Python-tulkki on vapaasti levitettävissä kaupalliseen ja ei-kaupalliseen käyttöön. Tulkki on saatavissa kaikille tavallisimmille laitteistoille ja käyttöjärjestelmille, joita ovat esimerkiksi Linux, Windows, Apple ja UNIX. Yhdessä ympäristössä kirjoitetut Python-ohjelmat toimivat yleensä muissa ilman muutoksia (poikkeuksena esimerkiksi käyttöjärjestelmän palveluita `os.system`-kutsun kautta käyttävät ohjelmat).

Kokonais- ja liukuluvut

Monien muiden tulkattujen ohjemointikielten tapaan Pythonia voi käyttää interaktiivisesti, eli eräänlaisena laskimena. Tällöin komennot syötetään suoraan Python-tulkin kehotteeseen. Python-tulkista poistuminen tapahtuu painamalla `CTRL+D`[1]. Interaktiivinen

[1] Tämä on UNIX-maailmasta tuttu tiedoston loppumiseen viittaava kontrollimerkki.

käyttötapa sopii hyvin kielen ominaisuuksiin tutustumiseen ja ohjelmoinnin harjoitteluun. Ohjeita jonkin käskyn tai kirjaston toiminnasta saa kirjoittamalle `help([käskyn nimi])`. Seuraavassa yksinkertainen esimerkki Python-tulkin interaktiivisesta käytöstä:

```
>>> 1+1
2
>>> 123+456
579
>>>
```

Lukujen esittämiseen tietokoneessa käytetään kokonaisluku- (Pythonissa `int`) ja liukulukuesityksiä (vast. `float`). Liukulukuja käytetään tavallisesti murto- ja irrationaalilukujen esittämiseen. Liukuluku on kuitenkin aina tarkkuudeltaan rajoitettu approksimaatio, joka vastaa likimäärin taskulaskimista tuttua luvun eksponenttiesitystä. Äärellinen esitystarkkuus aiheuttaa ongelmia, jos esimerkiksi lasketaan yhteen suuruusluokaltaan paljon toisistaan poikkeavia liukulukuja, kuten 10^{-10} ja 10^{10}. Erityisesti kahden liukuluvun yhtäsuuruuden vertaaminen suoraan ei yleensä johda toivottuun lopputulokseen. Tehtäessä laskutoimituksia kokonaisluvuilla lopputulos on kokonaisluku, liukuluvuilla vastaavasti liukuluku. Tämä voi aiheuttaa odottamattomia lopputuloksia. Liukulukuesitystä käytettäessä kokonaislukujen kokonaisosan jälkeen merkitään `.` (tai `.0`).

```
>>> 7/3
2
>>> 7./3.
2.3333333333333335
>>> 7./9.-(1./3.)*7./3
1.1102230246251565e-16
```

Koska `7` ja `3` käsitellään ensimmäisellä rivillä kokonaislukuina, jakolaskun lopputulos on myös kokonaisluku, eli osamäärän kokonaisosa. Liukulukujenkaan laskutoimituksen tuloksena ei äärellisen esitystarkkuuden vuoksi saada matemaattisesti oikeaa arvoa `2.3333...`. Tämä aiheuttaa ongelmia, jos halutaan esimerkiksi testata ovatko, kahden eri lausekkeen antamat arvot samat. Tällöin pitää samoiksi arvoiksi hyväksyä ne tapaukset, joissa arvot ovat riittävän lähellä toisiaan; käsitteen riittävän lähellä tulkinta riippuu sovelluksesta.

Muuttujista

Sijoittaminen muuttujaan tapahtuu sijoitusoperaattorin `=` avulla. Jos laskutoimituksen tai muun operaation lopputulos sijoitetaan muuttujaan, sitä ei tulosteta. Useampia muuttujia voi sijoittaa samanaikaisesti erotamalla muuttujat ja sijoitettavat arvot pilkulla.

```
>>> a,b=5,6
>>> a
5
>>> b
6
```

Jokaisella Python-muuttujalla (kuten muillakin lausekkeilla) on tyyppi, jonka voi selvittää komennolla `type`.

Merkkijonot ja listat

Kokonais- ja liukulukujen ohella muita hyödyllisiä muuttujia ovat esimerkiksi merkkijonot (`str`) ja listat (`list`). Merkkijonon merkkeihin osamerkkijonoihin voi viitata hakasulkujen `[]` avulla. Ilmaisu `[j:k]` tarkoittaa suljettua väliä j:stä $(k-1)$:een. Jos jompikumpi päätepiste puuttuu, tarkoitetaan väliä j:stä loppuun tai k:sta alkuun. Merkkijonon pituuden voi selvittää käskyllä `len`. Merkkijonojen (ja listojen) indeksointi Pythonissa alkaa 0:sta; indekseihin merkkijonon lopusta lukien viitataan negatiivisilla luvuilla. Merkkijonoja voi yhdistää operaation `+` avulla. Esimerkki merkkijonojen käytöstä:

```
>>> s='merkkijono'
>>> len(s)
10
>>> type(s)
<type 'str'>
>>> s[0]
'm'
>>> s[0:5]
'merkk'
>>> s[-5:]
'ijono'
>>> s[-1]
'o'
>>> t=' ja toinen merkkijono'
>>> s+t
'merkkijono ja toinen merkkijono'
```

Listat ovat indeksöityjä muuttujajonoja. Samaan listaan voidaan tallettaa erityyppisiä arvoja. Listan käsittely ja listan alkoihin viittaaminen muistuttaa merkkijonojen vastaavia operaatioita. Listan alkioilla on luonnollisesti omat tyyppinsä.

```
>>> li=['nolla','yksi','kaksi',3,4,5.0,6.0]
>>> li
['nolla', 'yksi', 'kaksi', 3, 4, 5.0, 6.0]
>>> li[1]
'yksi'
>>> li[2:4]
['kaksi', 3]
>>> len(li)
7
>>> type(li)
<type 'list'>
>>> type(li[2])
```

```
<type 'str'>
>>> type(li[6])
<type 'float'>
```

Tyyppimuunnokset

Muunnoksia erityyppisten muuttujien välillä voi tehdä tyyppimuunnosfunktioiden avulla. Niiden syntaksi on `tyyppi(·)`. Kaikkia tyyppejä ei voi luonnollisesti muuttaa toisikseen, ja lisäksi jotkin tyyppimuunnokset voivat hävittää informaatiota.

```
>>> s1='0.5'
>>> int(s1)
Traceback (most recent call last):
  File "<stdin>", line 1, in ?
ValueError: invalid literal for int(): 0.5
>>> float(s1)
0.5
>>> int(float(s1))
0
>>> float('1.0')
1.0
>>> str(1.0)
'1.0'
```

Tulostaminen `print`-käskyn avulla

Muuttujan arvon voi tulostaa käskyllä `print <muuttujan nimi>`. Käskyn yhteydessä on mahdollista antaa tulostuksen muodon määräävä merkkijono[2], joka erotetaan tulostettavista muuttujista prosenttimerkillä. Jos tulostettavia muuttujia on useita, niiden ympärille pitää merkitä sulkeet. Myös listan voi tulostaa `print`-käskyllä.

```
>>> a,b=5,6
>>> print a,b
5 6
>>> print '%f %f' % (a,b)
5.000000 6.000000
>>> print [a,b]
[5, 6]
```

Tulostuksen formatoinnin määrittämisessä voidaan käyttää mm. seuraavia optioita:

`%d` kokonaisluku,

`%f` liukuluku,

`%e` liukuluku, eksponenttiesitys,

`%s` merkkijono.

[2]Tämä vastaa C-kielen `printf`-käskyn toimintaa.

Tulostettavan merkkikentän pituuden ja mahdollisen esitystarkkuuden voi määrittää `print`-käskyn yhteydessä. Nämä annetaan pisteellä erotettuna prosenttimerkin jälkeen.

```
>>> a=1.23456789
>>> print "%10.6f" % a
  1.234568
>>> print "%10.3f" % a
     1.235
>>> print "%6.3f" % a
 1.235
>>> print "%6.3e" % a
1.235e+00
>>> print "%6.3d" % a
   001
```

Python-ohjelmat ja syötteen lukeminen käyttäjältä

Python-kielisten ohjelmatiedostojen tunnisteena käytetään tarkennetta `.py`. Ohjelmat ovat itsessään tavallisia tekstitiedostoja, ja niiden editoimiseen voi käyttää mitä hyvänsä ohjelmointiin soveltuvaa tekstieditoria, kuten esimerkiksi `notepad` tai `emacs`. Kommenttirivejä merkitään Pythonissa #-merkillä ja käskyn voi jakaa useammalle riville kenoviivan \ avulla. Kommenteissa olevat skandinaaviset merkit voivat aiheuttaa ongelmia joissakin järjestelmissä. Windowissa Python-tulkki asettaa itsensä automaattisesti `.py`-tiedostojen oletusarvoiseksi avausohjelmaksi. Ohjelmien ajaminen sujuu klikkaamalla ohjelmatiedoston kuvaketta. Ohjelmia voi ajaa myös komentoriviä käyttäen kirjoittamalla `python ohjelma.py`.

Pythonissa syötteen lukeminen käyttäjältä tapahtuu `input`-käskyllä. Käsky ottaa parametrinaan merkkijonon, joka on käyttäjälle syötettävä kehote.

```
>>> x=input('Anna luku: ')
Anna luku: 3
>>> print x
3
```

Käyttäjän antama syöte voi aiheuttaa myös ongelmia. Saattaa olla, että lukua pyydettäessä käyttäjä onkin antanut jonon kirjaimia. Tämän voi kuitenkin helposti estää tarkastamalla annetun syötteen tyyppi. Siihen tarvitaan kuitenkin seuraavaksi esitettäviä ehtorakenteita.

Ehto- ja toistorakenteet

Toistorakenteen `for` syntaksi poikkeaa Pythonissa useimmista muista ohjelmointikielistä. Toistolauseelle ei anneta parametriksi ehtoa, vaan lista, jonka jokaiselle alkiolle toisto suoritetaan. Toistettavaan haaraan liittyvät käskyt merkitään sisennyksen (tabuloinnin) avulla, eli `for [muuttuja] in [lista]: ....`

Jos asia halutaan toistaa k kertaa, voidaan käyttää käskyä `range(k)`, joka tuottaa listan, jossa ovat luvut $0, \ldots, k-1$. Parametrina voidaan antaa myös toinen luku j. Tässä tapauksessa `range(j,k)` tuotaa listan luvuista $j, \ldots, k-1$.

```
# Esimerkki: Lukujen toiset potenssit
for x in range(1,5):
        xx=x*x
        print '%d*%d = %d' % (x,x,xx)
```

```
Testiajo:

1*1 = 1
2*2 = 4
3*3 = 9
4*4 = 16
```

Lähes kaikista ohjelmointikielistä löytyvän `if-then-else` -rakenteen syntaksi Pythonissa on seuraava:
`if [ehto]: ...`
`elif [ehto2]: ...`
`else: ...`

Näistä `elif` ja `else` -haarat voi jättää pois. Haarassa `if` olevat käskyt suoritetaan, jos ehto toteutuu. Jos ehto ei toteudu, tutkitaan järjestyksessä `elif`-haarojen ehtoja. Jos mikään annetuista ehdoista ei toteutunut, suoritetaan `else`-haaran käskyt. Samaan haaraan kuuluvat käskyt merkitään `for`-lauseen tapaan sisennyksen avulla.

Tavallisimpia ehtoja ovat vertailut `<`, `<=`, `==`, `!=`, `>=` ja `>`. Huomaa, että yhtäsuuruutta verrattaessa käytetään merkintää `==`, jotta tehtäisiin ero sijoitusoperaation `=` kanssa[3]. Erisuuruudelle käytetään merkintää `!=`.

Toistorakennetta `while` käytetään, kun jotakin halutaan toistaa niin kauan, että jokin ehto ei enää ole voimassa. Tämä rakenne on erityisen hyödyllinen luettaessa syötettä käyttäjältä tai tiedostosta. Tällöin halutaan ehkä jatkaa lukemista, kunnes tiedosto on loppunut tai odottaa, että käyttäjä on antanut haluttujen rajojen puitteissa olevan syötteen. While-lauseen syntaksi on `while [ehto]: ....` Päättymättömän silmukan välttämiseksi täytyy toistettavan rakenteen sisällä luonnollisesti olla jotakin, mikä asettaa annetun ehdon epätodeksi, kun haluttu määrä toistoja on suoritettu.

```
# Esimerkki: Parilliset ja parittomat luvut
x='x'
# Syotetta kysytaan uudelleen, kunnes kayttaja
# antaa kokonaisluvun
while type(x)!=type(1):
        x=input('Anna kokonaisluku: ')
        if type(x)!=type(1): print x,' \
          ei ole kokonaisluku.\n'
if x%2==0: print "Luku %d on parillinen" % x
else: print "Luku %d on pariton" % x
```

```
Testiajo:

Anna kokonaisluku: 0.5
0.5  ei ole kokonaisluku.

Anna kokonaisluku: 4
Luku 4 on parillinen
```

Ohjelmassa esiintyvä merkintä `n%k` tarkoittaa n:n jakojäännöstä jaettaessa k:lla.

Kirjastojen käyttö

Monet hyödyllisistä Python-kielen ominaisuuksista on sijoitettu kirjastoihin, jotka täytyy ladata `import`-käskyllä ennen käyttöä. Kirjastoja ovat esimerkiksi matemaattisia funktioita sisältävä `math` ja käyttöjärjestelmään liittyviä käskyjä sisältävä kirjasto `os`. Kirjastoja voi helposti tehdä itsekin kirjoittamalla Python-tiedoston, joka sisältää pelkkiä funktioita. Kirjasto aktivoidaan käskyllä
`from [kirjasto] import [metodi]` tai
`import [kirjasto]`.
Ensimmäisessä tapauksessa metodit (funktiot) haetaan kirjastosta oletusnimiavaruuteen, eli metodeita voi kutsua suoraan kirjoittamalla `metodi([parametrit])`, jälkimmäisessä tapauksessa kirjaston metodeja pitää kutsua nimellä `[kirjasto].[metodi]([parametrit])`.

```
>>> from math import sin
>>> sin(3)
0.14112000805986721
>>> import math
>>> math.cos(3)
-0.98999249660044542
```

Jos kirjastosta halutaan hakea kaikki metodit, niin metodin paikalle voidaan merkitä `*`, siis esimerkiksi `from math import *`.

[3] Tämä ero on tärkeä, koska joissakin ohjelmointikielissä esimerkiksi ehto `if (x=1)` on aina tosi. Pythonissa sijoituslauseketta ei voi kirjoittaa ehtolauseeseen, esimerkiksi `if x=1: ...` antaa virheilmoituksen.

Funktioiden määrittely

Python-kielessä funktioiden määrittely tapahtuu `def`-rakenteen avulla. Käskyn syntaksi on
`def [funktion nimi]([parametrilista]):`.
Toistorakenteiden tapaan funktioon kuuluvat käskyt erotetaan sisennyksellä.

```
# Esimerkki: Toisen asteen yhtalon ratkaiseminen
from math import *

def solve2(a,b,c):
        D=b*b-4*a*c
# Liukulukua ei voi suoraan verrata nollaan
        if abs(a)<1.e-6:
                print "Virhe: Parametrin ",\
                      "a arvona ei saa olla 0.\n"
                return []
        elif abs(D)<1.e-6: return [-b/(2.*a)]
        elif D<0.: return []
        else: return [(-b+sqrt(D))/(2.*a), \
                      (-b-sqrt(D))/(2.*a)]
```

```
Testiajo:

>>> from esim3 import *
>>> solve2(1,0,0)
[0.0]
>>> solve2(1,0,1)
[]
>>> solve2(1,0,-1)
[1.0, -1.0]
```

Funktio `solve2` ratkaisee toisen asteen yhtälön reaaliset juuret. Funktion parametrit ovat kolme lukua a, b ja c, jotka vastaavat vakiokertoimia ratkaistavassa yhtälössä

$$ax^2 + bx + c = 0.$$

Funktio palauttaa listan yhtälön juurista. Jos reaalisia juuria ei ole, palautetaan tyhjä lista. Esimerkkiajon ensimmäinen tapaus vastaa yhtälöä $x^2 = 0$ (juuri 0:ssa), toinen $x^2 + 1 = 0$ (ei reaalisia juuria) ja kolmas $x^2 - 1 = 0$ (kaksi juurta, ± 1).

Satunnaisluvuista

Satunnaislukujen generoimiseen on Pythonissa käytössä kirjasto `whrandom`. Tästä kirjastosta löytyvä metodi `random` generoi tasaisesti jakautuneita satunnaislukuja (liukulukuja) puoliavoimelta väliltä $[0, 1)$. Vastaavasti metodi `uniform(a,b)` generoi satunnaislukuja väliltä $[a, b)$. Satunnaisten kokonaislukujen generoimiseen on käytössä metodi `randint(a,b)`, joka generoi satunnaisia kokonaislukuja väliltä $[a, b]$, eli mukaanlukien välin päätepisteet.

```
# Esimerkki: Nopanheiton simulointi
# Heitetaan noppaa 100 kertaa ja lasketaan
# silmalukujen jakauma
from whrandom import *
jak=[0,0,0,0,0,0]
for j in range(100):
        k=randint(1,6)
        jak[k-1]=jak[k-1]+1

for j in range(6): print '%d: %d'% (j+1,jak[j])
```

```
Testiajo:

1: 14
2: 15
3: 13
4: 18
5: 18
6: 22
```

Tiedostojen lukeminen ja kirjoittaminen

Jotta tiedostoa voitaisiin lukea tai siihen kirjoittaa, tiedosto on avattava `open`-metodilla. Käsittelyn jälkeen tiedosto suljetaan `close`-metodilla. Tiedoston jättäminen sulkematta voi aiheuttaa erilaisia virhetilanteita, kuten tiedostoon kirjoitetun datan menettämisen osittain tai kokonaan.

Käskyn `open` syntaksi on seuraava:
`[osoitin]=open('[tiedosto]', [moodi])`, missä `[tiedosto]` on avattavan tiedoston nimi ja `[moodi]` on joko `'r'` tai `'w'`, riippuen siitä, halutaanko tiedostoa lukea vai kirjoittaa siihen. Tiedostoon voidaan myöhemmin viitata nimen `[osoitin]` avulla. Tiedoston lukemiseen ja kirjoittamiseen on käytössä esimerkiksi `read`, `readlines` ja `write` -metodit.

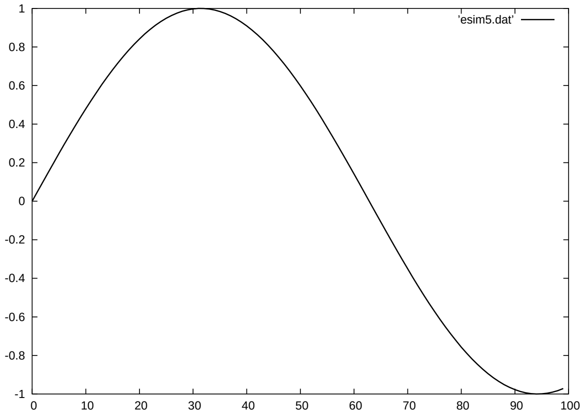
```
# Esimerkki: funktion arvojen kirjoittaminen
# tiedostoon
from math import *
import sys
dataf='esim5.dat'
fp=open(dataf,'w') # avataan tiedosto

# kirjoitetaan tiedotoon funktion sin(x) arvoja
for j in range(100):
        fp.write('%10.6f\n' % sin(0.05*j))
fp.close()          # suljetaan tiedosto
fp=open(dataf,'r')  # avataan tiedosto uudestaan
                    # lukemista varten
vals=fp.readlines() # luetaan tiedosto: tuloksena on
                    # lista merkkijonoja, vastaten
                    # tiedoston riveja
for v in vals: print '%10.6f' % float(v)
fp.close()          # suljetaan tiedosto
```

```
Testiajo:

  0.000000
  0.049979
  0.099833
  0.149438
  0.198669
  0.247404
...
```

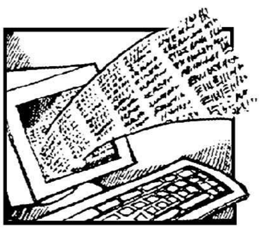
Funktion kuvaajan voi piirtää esimerkiksi käyttäen ilmaista Gnuplot-ohjelmaa[4]. Gnuplotille annettava käsky on `plot 'tiedosto.dat' w l`. Esimerkin tapauksessa saadaan seuraava kuva:



Linkkejä

- Pythonin kotisivu
  `http://www.python.org/`
- Aloittelijan opas Pythoniin – linkkejä alkuun pääsemiseksi
  `http://www.python.org/topics/learn/`
- Python Reference Card (pikaohje)
  `http://ourworld.compuserve.com/homepages/JasonRandHarper/PyQuickRef.pdf`
- Python FAQ (usein esitettyjä kysymyksiä)
  `http://www.python.org/doc/FAQ.html`
- Johdatus Pythoniin
  `http://archive.dstc.edu.au/python/python/Introduction.html`
- Guido van Rossumin (Pythonin isä) Python-opas
  `http://archive.dstc.edu.au/python/doc/tut/`
- Numerical Python
  `http://www.numpy.org`
- Gnuplotin kotisivu
  `http://www.gnuplot.info`

[4] Ohjeita Gnuplotin käyttöön löytyy sivulta `http://www.ucc.ie/gnuplot/gnuplot.html`.



Tietojenkäsittelytehtävien ratkaisuja

Solmun edellisessä numerossa 3/2003 oli matemaattisia tietojenkäsittelytehtäviä, joista toiseen ja kolmanteen esitetään tässä alkuperäiset KöMaLin ratkaisuehdotukset. Ensimmäiseen tehtävään ei KöMaLissa ole esitetty kuin lyhyt ratkaisuperiaate, eikä lukijoilta ole tullut ratkaisuja, joten niitä voi edelleen lähettää Solmun toimitukseen.

2. Binomikertoimet voidaan järjestää tavallisesta Pascalin kolmiosta oheisen kuvan osoittamalla tavalla. Lukuunottamatta kunkin rivin uloimpia alkioita jokainen luku on summa sen suoraan ja vasemmalla yläpuolella olevista luvuista.

1					$N = 6$	
1	*1*					
1	*2*	*1*				
1	*3*	*3*	*1*			
1	*4*	*6*	*4*	*1*		
1	*5*	*10*	*10*	*5*	*1*	
1	*6*	*15*	*20*	*15*	*6*	*1*

Tee Excel-taulukko, joka näyttää ensimmäiset $N + 1$ riviä tällä tavalla järjestetystä Pascalin kolmiosta. N:n arvo ($1 \leqslant N \leqslant 20$) syötetään ensimmäisen rivin seitsemänteen sarakkeeseen. Taulukon tulee aina sisältää tasan $N + 1$ riviä.

Ratkaisu. Ratkaisussa otetaan huomioon Pascalin kolmion hyvin tunnettu ominaisuus: jokainen alkio on yläpuolisten naapureidensa summa. Koska taulukkomme on vinossa, summataan suoraan yläpuolella ja vasemmalla yläpuolella olevat naapurit.

Ensimmäisessä sarakkeessa on

```
=IF(ROW()<=\$G\$1+1;1;""),
```

kaikkialla on siis joko ykkösiä tai ei mitään rivin numerosta riippuen.

Muissa soluissa on

```
=IF(ROW()<=\$G\$1+1;C5+D5;""),
```

sillä vain ne rivit, joiden rivinumero on korkeintaan $N + 1$ (N:n arvo on solussa `$G$1`), tulostetaan. Esimerkkikaava on solusta `D6`, ja siinä lasketaan yhteen solujen `D5` ja `C5` arvot.

Taulukon täyttäminen on yksinkertaista, koska Excel muuttaa automaattisesti solujen rivi- ja sarakeviittaukset kaavaa kopioitaessa.

3. Funktio Kertoma(N) $= N!$ kasvaa hyvin nopeasti. Vaikka $5! = 120$, niin jo luvun $10! = 3628800$ tallentamiseen tarvitaan 4-tavuinen kokonaisluku. Tietokone ei voi tallentaa lukua 100! 4- tai edes 8-tavuisena kokonaislukuna. Kuitenkin tiedetään, että jokainen luonnollinen luku voidaan hajottaa alkutekijöihin. Esimerkiksi $5! = 2^3 \cdot 3 \cdot 5$ ja $10! = 2^8 \cdot 3^4 \cdot 5^2 \cdot 7$. Tee ohjelma, joka lukee luvun $N (1 \leqslant N \leqslant 10000)$ näppäimistöltä ja tulostaa sen kertoman $N!$ alkutekijähajotelman.

Ratkaisu. Ratkaisemme ongelman antamalla kaksi algoritmiä, jotka tuottavat saman tuloksen.

Ratkaisu 1.

```
Vakio max_al = 5000;

Alkulukutek = Tietue
  lkm  : Luku
  elem : Taulukko( 1 .. max_al ) : Tietue
    alkuluku, eksp: Luku
  Tietueen loppu
Tietueen loppu

Muuttujat
  fakt, akt : Alkulukutek
  i, j, n   : Luku
```

Lue luku. Jos se on 1, tekijöinti on valmis, muussa tapauksessa aloitamme laskennan.

```
Lue: N (1<=N<=10000)
Jos N=1 Niin Tulosta: 1
Muuten
```

Ensimmäinen alkuluku on 2, eksponentilla 1.

```
  fakt.lkm:=1;
  fakt.elem[1].alkuluku:=2;
  fakt.elem[1].eksp:=1;
```

Seuraavaksi laskemme alkulukutekijöinnit luvuille 3:sta N:ään.

```
  Jokaiselle i:=3 .. N
    Alkulukutekijoi(i, akt)
```

Kunkin luvun tekijöinnistä syntyvät eksponentit lisätään jo saatuihin eksponentteihin.

```
    Jokaiselle i:=1 .. akt.lkm
      fakt.elem[j].eksp:=fakt.elem[j].eksp
      + akt.elem[j].eksp;
    Jokaiselle loppu
```

Jokainen uusi alkuluku lisätään listaan. (Tämä tapahtuu vain, jos uusi luku on alkuluku, joten kullakin askeleella alkulukujen määrä voi kasvaa korkeintaan yhdellä.)

```
    Jos fakt.lkm < akt.lkm Niin
      fakt.lkm:=fakt.lkm + 1;
      fakt.elem[fakt.lkm]:=akt.elem[akt.lkm]
    Jos loppu
  Jokaiselle loppu
```

Lopuksi tulostamme kaikki kertoman tekijöinnin elementit.

```
  Jokaiselle i:=1 .. fakt.lkm
    Tulosta: fakt.elem[i].alkuluku,
    fakt.elem[i].eksp;
Jos loppu

Proseduuri Alkulukutekijoi(i : luku; akt :
alkulukutek)
```

Käytämme funktiota `Alkuluku` päättämään, onko luku alkuluku vai ei.

```
  Funktio Alkuluku(j : luku) : looginen
```

Etsimme jakajia luvusta 2 luvun neliöjuureen.

```
    Toista niin kauan kun (k*k<j) Ja
    ((j mod k)>0)
      k:=k+1
    Toista loppu
```

Jos i jakaantuu, se ei ole alkuluku, muussa tapauksessa se on.

```
    Alkuluku:=(k*k>j)
  Funktio loppu
```

2 on ensimmäinen alkuluku, eksponentilla 1.

```
  j:=2;
  akt.lkm:=1;
  akt.elem[j].alkuluku:=2;
  akt.elem[j].eksp:=0;
```

Jatkamme niin kauan kun alkuluku on vähemmän tai yhtäsuuri kuin i.

```
  Toista niin kauan kun j<=i
```

Jos j jakaa i:n, eksponenttia kasvatetaan.

```
    Jos i mod j = 0 Niin
      akt.elem[akt.lkm].eksp:=
      akt.elem[akt.lkm].eksp + 1;
      i:=i div j;
    Muuten
      j:=j+1;
```

Jos jako ei mene tasan, etsimme seuraavan (i:tä pienemmän) alkulukujakajan.

```
      Toista niin kauan kun (j<=i) Ja
      ei Alkuluku(j)
        j:=j+1;
      Toista loppu
```

Jos löydetään alkuluku, saamme uuden alkuluvun j, eksponentilla 0. (Eksponentti kasvatetaan seuraavan kierroksen alussa yhteen).

```
      Jos j<=i Niin
        akt.lkm:=akt.lkm + 1;
        akt.elem[akt.lkm].alkuluku:=j;
        akt.elem[akt.lkm].eksp:=0;
      Jos loppu
    Jos loppu
  Toista lopppu
```

Viimeisellä nollaeksponentilla ei ole merkitystä.

```
  Jos akt.elem[akt.lkm].eksp = 0 Niin
  akt.lkm:=akt.lkm - 1;
Proseduuri loppu
```

Ratkaisu 2. Käytämme Legendren kaavaa seuraavassa muodossa. Jos p on mielivaltainen alkuluku ja N positiivinen kokonaisluku, niin p:n eksponentti $N!$:n alkulukutekijöinnissä saadaan kaavalla

$$\left\lfloor \frac{N}{p} \right\rfloor + \left\lfloor \frac{N}{p^2} \right\rfloor + \left\lfloor \frac{N}{p^3} \right\rfloor + \dots$$

Summa on äärellinen, koska kokonaislukuosat ovat nolla kun $N < p^i$.

Luetaan luku.

```
Lue: N
```

Jos luku on 1, tekijöinti on valmis.

```
Jos N=1 Niin Tulosta: 1
```

Pienin alkuluku on kaksi.

```
p:=2;
```

Jos 2 ei ole suurempi kuin N, 2 eksponentteineen tulostetaan.

```
Jos p<=N Niin Tulosta: p, Eksponentti(p,N);
```

Seuraava alkuluku on 3.

```
p:=3;
```

Niin kauan kun alkuluku ei ole suurempi kuin luku, alkuluku ja sen eksponentti tulostetaan.

```
Toista niin kauan kun p<=N
  Tulosta: p, Eksponentti(p,N);
  Seuraava_alkuluku(p);
Toista loppu
```

Alkuluvun eksponentin laskeminen Legendren kaavalla.

```
Funktio Eksponentti(p,N : luku) : luku
```

Eksponentti on laskettava summa, aluksi nolla. pp:ssä on p:n potenssit, aluksi p.

```
  Eksponentti:=0;
  pp:=p;
```

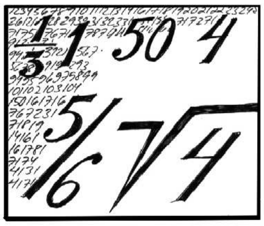
Jatkamme niin kauan kuin p:n potenssi on vähemmän tai yhtäsuuri kuin N.

```
  Toista niin kauan kun pp<=N
```

Eksponenttia kasvatetaan ja seuraava potenssi lasketaan.

```
    Eksponentti:=Eksponentti +
    Pyoristys(n/pp);
    pp:=pp * p;
  Toista loppu
Funktio loppu
```

Lähde: *KöMaL*, Volume 1, Number 1, December 2002, 66–69.
Käännös ja ladonta: ***Timo Metsälä***



1089 ja muita matemaattisia yllätyksiä

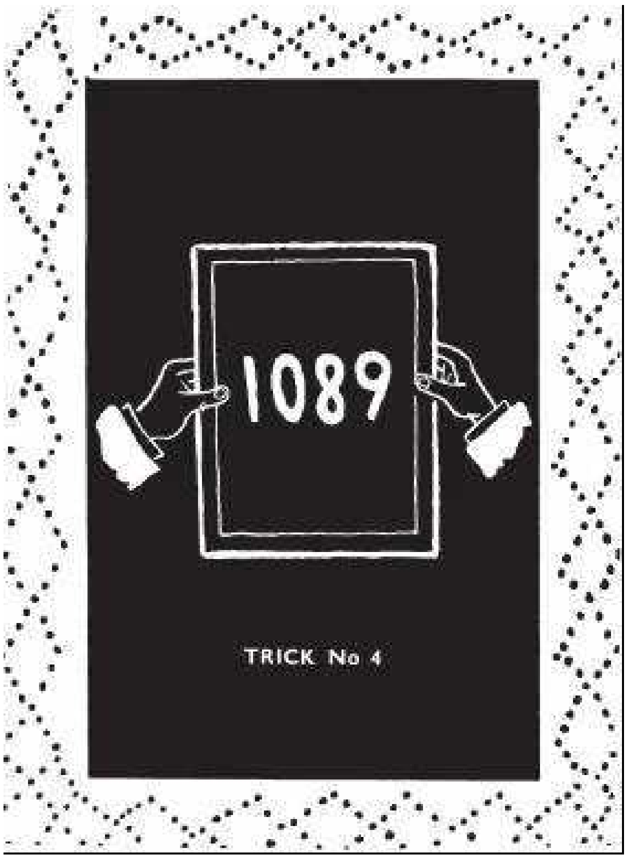
David Acheson
Matematiikan professori
Jesus College, Oxfordin yliopisto, Iso-Britannia

Miksi niin monet ihmiset sanovat inhoavansa matematiikkaa? Aivan liian usein totuus on, että heidät on pidetty loitolla todellisesta matematiikasta, ja uskon, että matemaatikot voisivat halutessaan ponnistella enemmän tuodakseen alansa pieniä ilonaiheita ja ajatuksenpoikasia suuren yleisön tietoisuuteen. Eräs tapa tehdä tämä olisi ehkä korostaa yllätyksellisyyttä, joka on usein läsnä matematiikassa parhaimmillaan. Kaikkihan pitävät iloisista yllätyksistä!

Numerotemppu

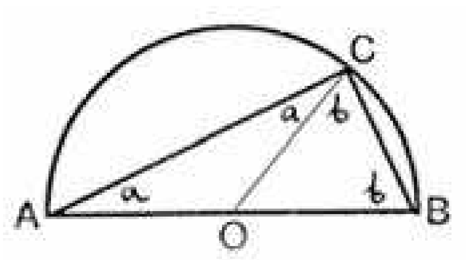
Koin ensimmäisen matemaattisen yllätykseni vuonna 1956, ollessani kymmenen vanha. Olin tuolloin innostunut taikatempuista, ja yhtenä päivänä löysin sattumalta "numerotempun" eräästä artikkelista nimeltä *Uncle Jack turns you into a Conjuror* (conjuror = taikuri).

Temppu tehdään näin: Ota mikä tahansa kolminumeroinen luku, jossa ensimmäisen ja viimeisen numeron erotus on vähintään kaksi. Käännä luku toisin päin, ja vähennä suuremmasta luvusta pienempi (esim. $782 - 287 = 495$). Käännä sitten saamasi erotus ja laske näin saatu luku yhteen kyseisen erotuksen kanssa ($594 + 495 = 1089$). Merkillistä tässä on se, että tulos on aina 1089 riippumatta alussa valitusta kolminumeroisesta luvusta. Tänä päivänä olen tietenkin selvillä siitä, että tämä 1089-temppu on matemaattisessa mielessä melko harmiton höyhensarjalainen. Mutta jos siihen sattuu törmäämään 10-vuotias miehenalku vuonna 1956, se saattaa kyllä panna sukat pyörimään jalassa.



Ällistyttävää geometriaa

Hiukan myöhemmin pääsi geometria yllättämään minut ensimmäistä kertaa. Eräänä päivänä meille kerrottiin koulussa, että jos AB on ympyrän halkaisija ja C on mikä tahansa ympyrän kehän piste, niin kulma ACB on suora. Muistelen, että tätä oli jokseenkin vaikea uskoa: minusta näytti siltä, että jos pistettä C siirrettäisiin ympyrän kehää pitkin, niin kulma ACB melko varmasti muuttuisi – erityisesti C:n lähestyessä B:tä. Sitten esitettiin kuitenkin todistus, että näin ei ole. Kas näin:



Piirrä ensin suora viiva pisteestä C ympyrän keskipisteeseen O. Tällöin $OA = OB = OC$. Näin ollen kolmio AOC on tasakylkinen, ja kuvassa a:lla merkityt kulmat ovat siis yhtäsuuret. Vastaavasti perustellaan b:llä merkittyjen kulmien yhtäsuuruus kolmiossa BOC. Koska kolmion ACB kulmien summa on 180°, niin $a + (a + b) + b = 180°$. Siis $a + b = 90°$, ja kulma ACB on täten suora. Yhä vieläkin tämä todistus on mielestäni yksi tehokkaimmista ja paljastavimmista koko matematiikassa.

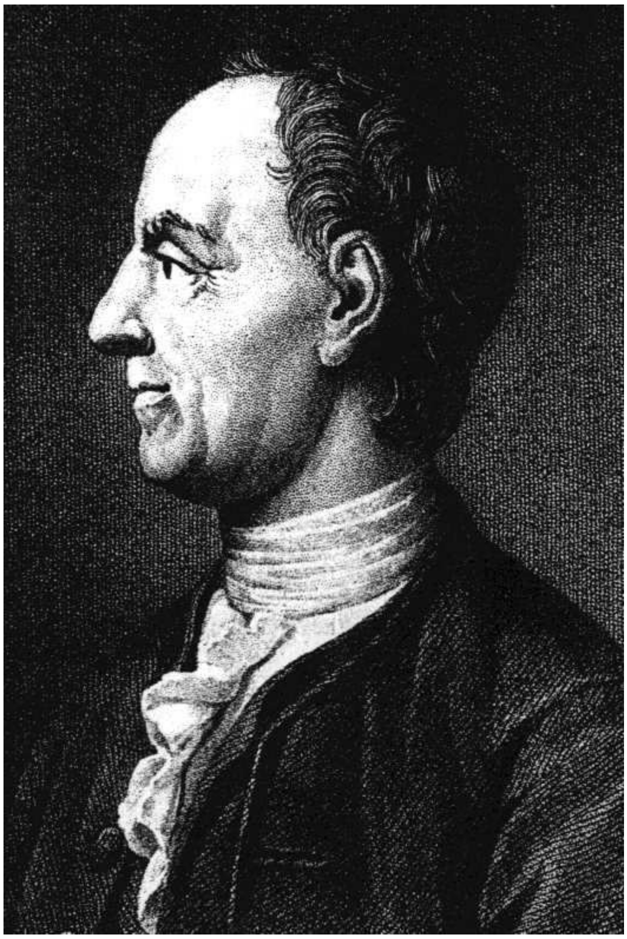
Suuria erehdyksiä

Koko todistamisen idea itsessään on ehkä yksi suurimmista hankaluuksista välitettäessä matematiikkaa laajemmalle yleisölle. Me matemaatikot saatamme helposti vaikuttaa tolkuttoman innostuneilta omasta alastamme, ja mielestäni meidän pitää olla valmiita selittämään muillekin, mikä siinä on niin tärkeää. Voisimme vaikkapa korostaa sitä, että todistusten puuttuessa matemaatikot saattavat mennä pahasti metsään, eivätkä parhaatkaan matemaatikot ole virheille immuuneja.

Euler esimerkiksi todisti vuonna 1753 Fermat'n viimeisen lauseen tapauksessa $n = 3$. Hän toisin sanoen osoitti, että yhtälölle

$$a^3 + b^3 = c^3$$

ei löydy kokonaislukuratkaisuja – siis kahden kuution summa ei voi olla kuutio. Hieman myöhemmin hän laajensi epäilyään koskemaan neljättä potenssia: yhtä lailla mahdotonta olisi, että kolmen neljännessä potenssissa olevan kokonaisluvun summa olisi neljännessä potenssissa. Ja vielä yleisemmin, hän arveli, että olisi mahdotonta laskea yhteen $m - 1$ kappaletta potenssiin m korotettuja kokonaislukuja siten, että summakin olisi m:nnessä potenssissa. Lähes kahteensataan vuoteen kukaan ei löytänyt mitään vikaa tästä väittämästä. Kukaan ei varsinaisesti osannut todistaa sitä, mutta vastaesimerkitkin loistivat poissaolollaan. Ja loppujen lopuksi, olihan kyseisen veikkauksen esittänyt erittäin arvovaltainen auktoriteetti.



Leonhard Euler (1707–83)

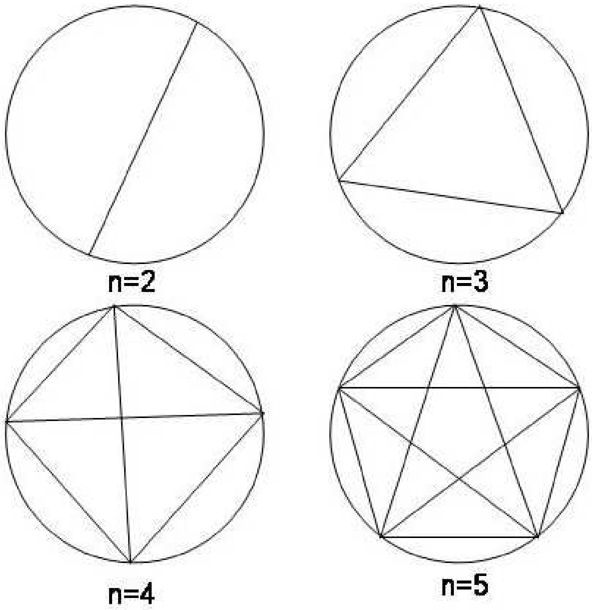
Sitten, vuonna 1966, *L.J. Lander* ja *T.R. Parkin* keksivät vastaesimerkin – neljä kappaletta 5:nnessä potenssissa olevia lukuja, joiden summa oli samassa potenssissa:

$$27^5 + 84^5 + 110^5 + 133^5 = 144^5.$$

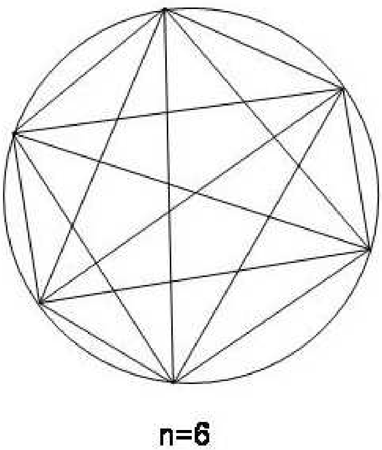
20 vuotta tämän jälkeen *N. Elkies* vuorostaan kumosi tapauksen $m = 4$:

$$2682440^4 + 15365639^4 + 18796760^4 = 20615673^4.$$

Yhden–kahden erikoistapauksen perusteella yleistäminen on matematiikassa tietenkin aina riskibisnestä. Eräs mielestäni osuvimmista esimerkeistä yleistämisen vaarallisuudesta liittyy näennäisen pieneen ja viattomaan geometriseen ongelmaan. Piirrä ympyrä, merkitse sen kehälle n pistettä ja yhdistä pisteet toisiinsa janoilla. Ne jakavat ympyrän osiin, ja kysymys kuuluukin: kuinka moneen? (Oletetaan että kussakin ympyrän sisäpisteessä leikkaa korkeintaan kaksi janaa.)



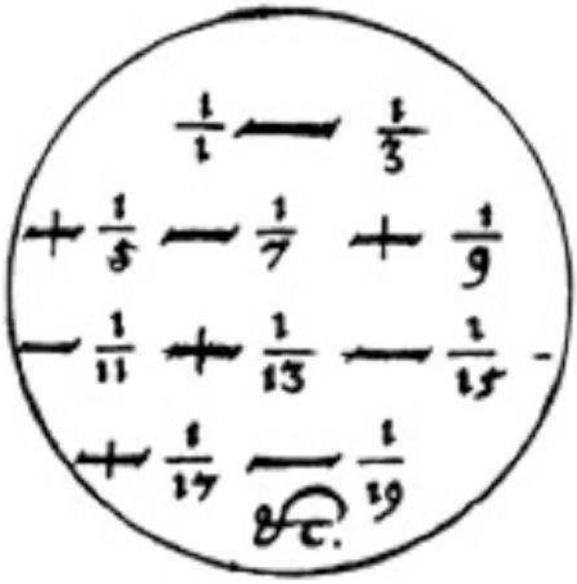
Muutamalla ensimmäisellä n:n arvolla – 2, 3, 4 ja 5 – alueiden lukumäärä käyttäytyy yksinkertaisen säännönmukaisesti: 2, 4, 8, 16. Ja ainakin omien kokemusteni mukaan tässä vaiheessa on mahdollista houkutella melkein kenet tahansa "päättelemään", että n:n arvolla 6 alueiden lukumäärä on 32. Näin ei kuitenkaan ole, vaan niitä on 31:



Yleinen kaava alueiden lukumäärälle ei siis ole $2^n - 1$, vaan

$$\frac{n^4 - 6n^3 + 23n^2 - 18n + 24}{24}.$$


Numero DEUS
I
impare gaudet


$$\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \frac{1}{13} - \frac{1}{15} + \frac{1}{17} - \frac{1}{19}$$ &c.

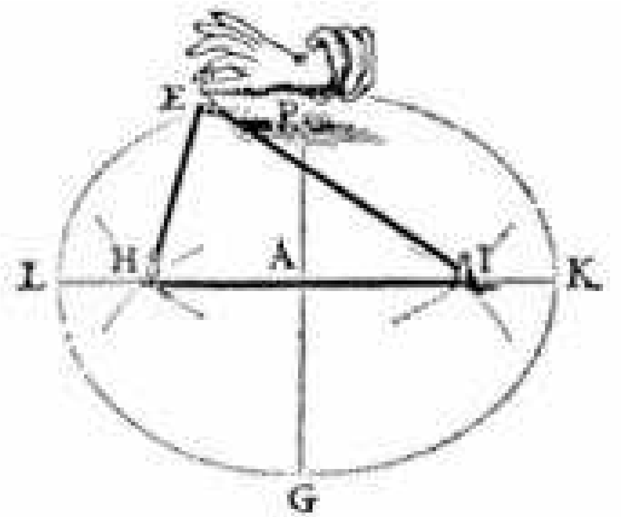
Kuvitusta Leibnizin artikkelista vuodelta 1674.

Yllättäviä yhteyksiä

Korkeammassa matematiikassa eräät suurimmista kummallisuuksista ilmenevät odottamattomina yhteyksinä eri osa-alueiden välillä. Opiskeltuamme jonkin verran esimerkiksi differentiaali- ja integraalilaskentaa törmäämme kuuluisaan Gregory–Leibniz -sarjaan

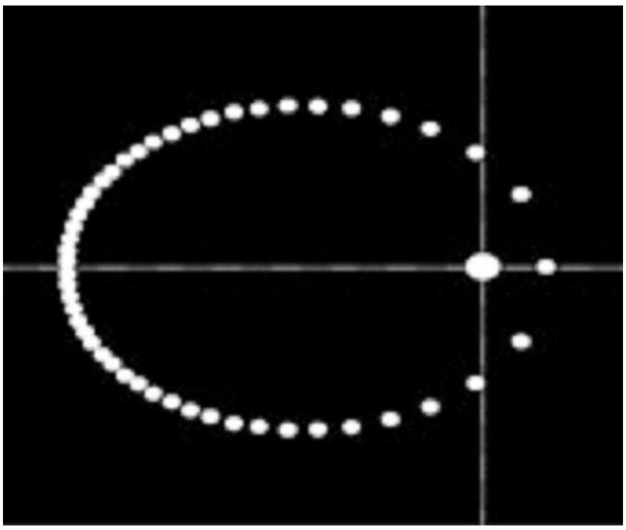
$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots$$

Tarkkakaan todistus ei minun silmissäni paljoakaan vähennä sitä hämmästystä ja kummastusta, jonka tämä erikoislaatuinen tulos saa aikaan. Kun nimittäin tapaamme π:n ensimmäistä kertaa, se yhdistetään vain ympyröihin, enkä ole tähän mennessä tavannut ketään, joka osaisi yksinkertaisella tavalla selittää ympyröiden ja parittomien lukujen käänteislukujen välisen yhteyden.



Ellipsi van Schootenin teoksesta Exercitationum mathematicorum (1657).

Aina yllättävät yhteydet eivät kuitenkaan löydy matematiikasta itsestään, vaan matematiikan ja reaalimaailman väliltä. Esimerkiksi ellipsi on jo kreikkalaisten matemaatikoiden hyvin tuntema käyrä, joka voidaan konstruoida kuvan osoittamalla tavalla. Pisteitä H ja I kutsutaan polttopisteiksi.



Ensi silmäyksellä tämä saattaa näyttää ”vain” geometrialta. Jätetään kuitenkin hetkeksi puhdas geometria ja ajatellaan esimerkiksi planeettaa (tai komeettaa) pistemäisenä massana, ja oletetaan, että sen ja jonkin kiinteän ”auringon” välillä vaikuttaa etäisyyden neliölakia noudattava vetovoima. Ratkaistuamme asiaankuuluvat differentiaaliyhtälöt havaitsemme, että kaikkien suljettujen kiertoratojen täytyy olla ellipsejä, ja lisäksi aurinko on aina yhdessä polttopisteistä!

Melkein kuin intialainen köysitemppu

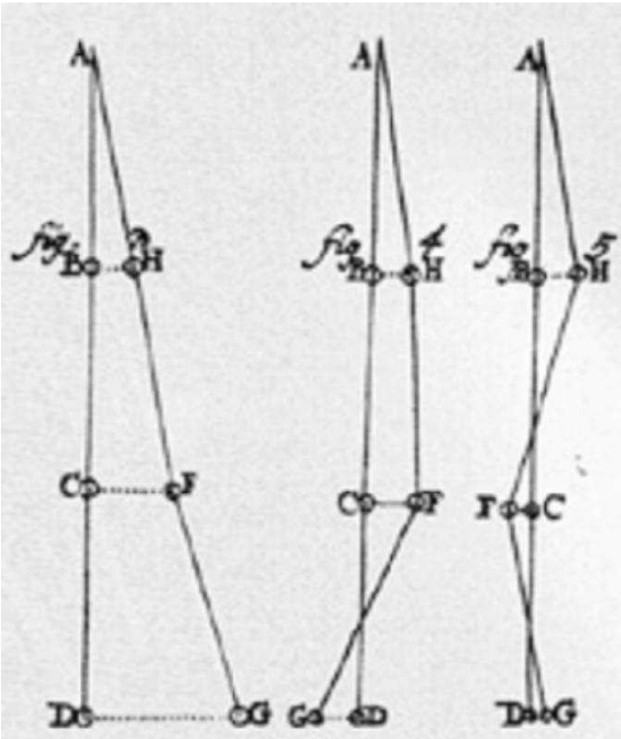
Koin elämäni luultavasti suurimman matemaattisen yllätyksen eräänä sateisena marraskuun iltapäivänä 1992, jolloin löysin itseni todistamasta omituista uutta teoreemaa. Yritin saada uutta vääntöä vanhaan dynamiikkaan liittyvään ongelmaan, jota ensimmäisenä oli tutkinut *Daniel Bernoulli* vuonna 1738. Hän oli tutkinut N:n toisiinsa yhdistetyn heilurin ketjua ja löytänyt N erilaista heilahtelutapaa. Alimmalla tasolla, taajuuden ollessa f_1, heilurit heilahtelevat yhdessä edestakaisin, melkein kuin muodostaen yhden pitkän heilurin. Korkeimmalla taajuustasolla f_N peräkkäiset heilurit heilahtelevat joka hetki vastakkaisiin suuntiin.

Daniel Bernoulli

Minun teoreemani osoitti, että N:n heilurin ketju voidaan kääntää ylösalaisin niin, että ne ovat horjuvat jotenkuten tasapainossa toinen toisensa päällä, ja sitten vakauttaa ne kyseisessä asennossa panemalla alimman heilurin tukipiste värähtelemään pystysuunnassa. Jos f_N^2 on paljon suurempi kuin f_1^2 (kuten yleensä on), tasapainoehdoiksi osoittautuvat

$$a < \frac{0.0114g}{f_N^2} \text{ ja } af_p > \frac{g}{2\sqrt{2}\pi^2 f_1},$$

missä a tarkoittaa tukipisteen värähtelysädettä, f_p tukipisteen värähtelytaajuutta ja g maan vetovoiman kiihtyvyyttä. Näin teoreema riippuu hyvin yksinkertaisella tavalla vain kahdesta luvusta f_1 ja f_N, jotka määräävät kuinka heilurit värähtelevät tukipisteen yläpuolella. Lopputuloksena on, että "temppu" voidaan tehdä aina kun tukipiste saadaan värähtelemään pystysuunnassa niin, että värähtelysäde on tarpeeksi pieni ja taajuus riittävän korkea.



Heilurikolmikon kolme värähtelytapaa, Daniel Bernoullin alkuperäisestä artikkelista vuodelta 1738.

Pian sen jälkeen, kun aloimme kutsua aihetta koskevia esitelmiämme nimellä "Melkein kuin intialainen köysitemppu", huomasimme että sanomalehdet, radio ja televisio alkoivat kaikki kiinnostua asiasta, ja aiheesta on ollut minulle ja Tomille vuosien kuluessa paljon hupia. Jopa lontoolainen Magic Circle on hankkinut asiaa koskevat tieteelliset julkaisuni arkistoihinsa, mikä olisi varmasti hämmästyttänyt erästä 10-vuotiasta poikaa vuonna 1956. (Papereita säilytetään tietääkseni mapissa, jonka nimi on *Sundry Ephemera*, vap. suom. Kaikenlaista vessapaperia.)

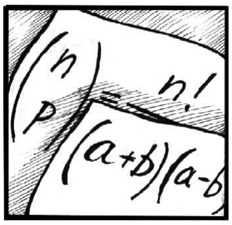
Loppujen lopuksi ei kuitenkaan ole kovin oleellista, voidaanko jokin taikatemppu selittää matematiikan avulla vai ei. Merkitystä sen sijaan varmasti on sillä, missä määrin tämänkaltaiset yllättävät lopputulokset voivat saada suuren yleisön vakuuttuneeksi matematiikan taianomaisuudesta.

Lähde

David Acheson, *1089 and all that: A journey into mathematics*, Oxford University Press, 2002. (Erityisesti luku 15 antaa lisävalaistusta "köysitemppuun".)

Tämä artikkeli on lyhennelmä kirjoittajansa London Mathematical Societyn yleisöluennosta "Mathematics, magic and the electric guitar". Achesonin kirja *1089 and all that* on omaperäinen yritys tarjoilla matemaattisia herkkupaloja keskivertokansalaisille. Lisää tietoa aiheesta löytyy osoitteesta www.jesus.ox.ac.uk/$\sim$dacheson.

Alkuperäisartikkeli: David Acheson, 1089 and all that: The elements of surprise in mathematics, *EMS Newsletter*, numero 49, syyskuu 2003, s. 9–11. Artikkelin kääntämiseen ja Solmussa julkaisuun on saatu lupa David Achesonilta ja EMS:n lehdeltä. Käännös ja ladonta: ***Anna Eteläaho***



Yksinkertaisista jaollisuustesteistä

Timo Tossavainen
Lehtori
Savonlinnan OKL, Joensuun yliopisto

Lukion pitkässä matematiikassa jaollisuustestejä käsitellään ainakin jossakin laajuudessa logiikan ja lukuteorian syventävällä kurssilla. Solmussa jaollisuustestien kannalta keskeistä kongruenssin käsitettä on puolestaan tarkasteltu ainakin artikkeleissa [2] ja [5]. Tämä kirjoitus on jonkinlainen yhteenveto siitä matematiikasta, jota yksinkertaisten jaollisuustestien konstruoimiseksi tarvitaan.

Jakoyhtälö ja kongruenssi

Jakoalgoritmin tai oikeammin jakoyhtälön nimellä tunnettu lause on eräs keskeisimmistä lukuteorian työkaluista. Se voidaan ilmaista esimerkiksi seuraavassa muodossa.

Lause 1. Olkoot a ja n kokonaislukuja siten, että $n > 0$. Tällöin on olemassa yksikäsitteiset kokonaisluvut q ja r siten, että

$$a = qn + r \quad \text{ja} \quad 0 \leq r < n.$$

Todistus. Olkoon

$$E = \{r \in \mathbb{Z} : r \geq 0 \text{ ja } r = a - qn \text{ jollakin } q \in \mathbb{Z}\}.$$

Tällöin E on epätyhjä, sillä jos $a \geq 0$, on $a \in E$, ja jos $a < 0$, on $a - an \in E$. Näin ollen joukossa E on pienin alkio $r_0 = a - q_0 n$. Lisäksi $r_0 < n$, sillä muuten

$$r_1 = a - (q_0 + 1)n = r_0 - n \geq 0$$

ja siksi $r_1 \in E$, mikä olisi ristiriita.

Oletetaan, että

$$a = qn + r = q'n + r', \text{ missä } 0 \leq r < n \text{ ja } 0 \leq r' < n.$$

Tällöin

$$r' - r = (q - q')n,$$

joten $r' - r$ on jaollinen luvulla n. Koska $0 \leq r < n$ ja $0 \leq r' < n$, on

$$-n < r' - r < n.$$

Näin ollen $r' - r = 0$ eli $r = r'$, jolloin myös $q = q'$, koska $n > 0$. $\square$

Usein sanotaan, että luku a on *jaettava*, n *jakaja*, q *osamäärä* ja r *jakojäännös*. Jos luvuilla a ja b on yhteisen jakajan n suhteen samat jakojäännökset, tällöin luvut a ja b ovat *kongruentit modulo* n, ja tätä merkitään kirjoittamalla

$$a \equiv b \pmod{n}.$$

Määritelmästä seuraa välittömästi, että luvut a ja b ovat kongruentit modulo n, jos ja vain jos $a - b$ on jaollinen luvulla n.

Esimerkki. Koska $17 = 3 \cdot 5 + 2$ ja $-18 = -4 \cdot 5 + 2$, niin

$$17 \equiv -18 \pmod 5.$$

Toisaalta sama asia nähdään siitä, että $17 - (-18) = 35 = 7 \cdot 5$.

Kongruenssi modulo n on ekvivalenssirelaatio kokonaislukujen joukossa (ks. esim. [5]), joten se toimii ikäänkuin relaatio $=$ kokonaislukujen tavallisessa aritmetiikassa. Jakojäännösten aritmetiikka muistuttaa muutenkin monessa suhteessa kokonaislukujen tavallista aritmetiikkaa, sillä voimme todistaa muun muassa seuraavat kongruentteja lukuja koskevat laskusäännöt.

Lause 2. Olkoon $a \equiv b \pmod n$ ja $c \equiv d \pmod n$. Tällöin

$$a + c \equiv b + d \pmod n$$

ja

$$ac \equiv bd \pmod n.$$

Todistus. Koska $a - b = kn$ ja $c - d = ln$ joillakin $k, l \in \mathbb{Z}$, on

$$(a + c) - (b + d) = (a - b) + (c - d) = (k + l)n.$$

Lukujen $a + c$ ja $b + d$ erotus on siis jaollinen luvulla n, joten ne ovat kongruentit modulo n.

Vastaavasti

$$\begin{aligned} ac - bd &= (ac - bc) + (bc - bd) \\ &= (a - b)c + (c - d)b \\ &= (ck + bl)n, \end{aligned}$$

mistä toinen väite seuraa. □

Olkoon $k \geq 2$ luonnollinen luku. Induktion avulla tai soveltamalla Lausetta 2 riittävän monta kertaa peräkkäin näemme edelleen, että

$$\sum_{i=1}^{k} a_i \equiv \sum_{i=1}^{k} b_i \pmod n \qquad (1)$$

ja

$$a_1 a_2 \cdots a_k \equiv b_1 b_2 \cdots b_k \pmod n \qquad (2)$$

jos $a_i \equiv b_i \pmod n$ kaikilla $i = 1, ..., k$.

Huomautus. Jakojäännösten aritmetiikka poikkeaa kuitenkin hieman kokonaislukujen tavallisesta aritmetiikasta. Koska esimerkiksi $14 \equiv 8 \pmod 6$ ja $7 \not\equiv 4 \pmod 6$, kongruenssiyhtälön puolittainen jakaminen ei onnistu kaikilla vakioilla.

Kongruenssi ja jaollisuustestit

Se, onko luku x jaollinen luvulla n, selviää yleensä helpoiten jakamalla x luvulla n esimerkiksi taskulaskimen avulla tai käsin jakokulmassa. Jos x on hyvin suuri, jakaminen ei kuitenkaan onnistu taskulaskimella. Toisaalta jakokulmassa laskemistakaan ei voida pitää erityisen tehokkaana tapana tutkia suurten lukujen jaollisuutta. Tällöin on etsittävä joko uusia tapoja suorittaa jakolasku tai sitten menetelmiä, joissa testattava luku voidaan korvata sellaisella luvulla, jonka jakaminen onnistuu helpommin. Lause 2 ja kaavat (1) ja (2) osoittautuvat tässä hyödyllisiksi.

Koska kokonaisluvun x jaollisuus luvulla $n > 0$ ei riipu sen etumerkistä, riittää tarkastella vain positiivisten kokonaislukujen jaollisuutta. Tällöin jokaista lukua x vastaa luonnollinen luku $k \geq 1$ siten, että

$$x = a_k 10^k + a_{k-1} 10^{k-1} + ... + a_1 10 + a_0, \qquad (3)$$

missä $a_i \in \{0, 1, 2, ..., 9\}$ kaikilla $i = 0, 1, ..., k$. Jakoyhtälön nojalla kaikilla $i \in \mathbb{N}$ löytyy luku b_i siten, että $0 \leq b_i < n$ ja 10^i on kongruentti luvun b_i kanssa modulo n. Tällöin Lauseen 2 ja kaavojen (1) ja (2) nojalla on x jaollinen luvulla n, jos ja vain jos

$$a_k b_k + a_{k-1} b_{k-1} + ... + a_1 b_1 + a_0 \equiv 0 \pmod n. \qquad (4)$$

Jos x on suuri, on kaavan (4) vasen puoli huomattavasti pienempi kuin x. Lisäksi kaavaa (4) voidaan soveltaa useita kertoja peräkkäin.

Kolmella jaollisuus

Olkoon $n = 3$. Koska $10 \equiv 1 \pmod 3$, on kaavan (2) nojalla $10^k \equiv 1 \pmod 3$ kaikilla $k \geq 1$. Luvut b_i ovat siis kaikki ykkösiä kaavassa (4), joten kaavan (3) muodossa esitetty luku x on jaollinen luvulla 3, jos ja vain jos

$$a_k + a_{k-1} + ... + a_1 + a_0 \equiv 0 \pmod 3.$$

Esimerkiksi 29 760 183 on jaollinen luvulla kolme, sillä

$$2 + 9 + 7 + 6 + 0 + 1 + 8 + 3 \equiv 36 \equiv 0 \pmod 3.$$

Kahdella ja viidellä jaollisuus

Olkoon $n \in \{2, 5\}$. Koska $10 \equiv 0 \pmod n$, on kaavan (2) nojalla $10^k \equiv 0 \pmod n$ kaikilla $k \geq 1$. Tällöin kaavan (3) muodossa esitetty luku x on jaollinen luvulla n, jos ja vain jos

$$a_0 \equiv 0 \pmod n.$$

Esimerkiksi 976 213 521 ei ole jaollinen kahdella eikä viidellä, sillä

$$1 \not\equiv 0 \pmod n,$$

silloin kun $n \in \{2, 5\}$.

Joissakin jaollisuustesteissä kannattaa sallia luvuille b_i myös negatiivisia arvoja. Jos x ei ole luvun n monikerta, jakoyhtälöstä seuraa, kun alkuperäinen osamäärä korvataan yhtä suuremmalla kokonaisluvulla, että x voidaan kirjoittaa yksikäsitteisesti myös muodossa

$$x = qn + r, \quad \text{missä} \quad q \in \mathbb{Z} \text{ ja } -n < r < 0.$$

Jokainen kokonaisluku, ja erityisesti jokainen 10^i, on siis kongruentti myös sellaisen luvun b_i kanssa, missä $-n < b_i \leq 0$.

Jaollisuus luvulla 11

Koska $10 \equiv -1 \pmod{11}$, kaavasta (2) seuraa, että $10^k \equiv (-1)^k \pmod{11}$ kaikilla $k \geq 1$. Näin ollen kaavan (3) muodossa esitetty luku x on jaollinen luvulla 11, jos ja vain jos

$$a_k(-1)^k + a_{k-1}(-1)^{k-1} + ... - a_1 + a_0 \equiv 0 \pmod{11}.$$

Edellisen kaavan nojalla on siis ilmeistä, että esimerkiksi 987 654 321 123 456 789 on jaollinen luvulla 11.

Joskus testattava luku x kannattaa esittää muodossa, jossa kantalukuna on 100, 1000 tai vieläkin suurempi kymmenen potenssi. (Yksinkertaisimmat kertoimet luvulla n jaollisuuden testiin saadaan tietysti silloin, kun luku x esitetään n-lukujärjestelmän mukaisessa muodossa...)

Jaollisuus luvulla 13

Koska $10 \equiv -3 \pmod{13}$, $10^2 \equiv -4 \pmod{13}$, $10^3 \equiv -1 \pmod{13}$, $10^4 \equiv 3 \pmod{13}$, $10^5 \equiv 4 \pmod{13}$, $10^6 \equiv 1 \pmod{13}$, $10^7 \equiv -3 \pmod{13}$ jne., on arvolla $n = 13$ kaavassa (4) kertoimet $b_1 = -3$, $b_2 = -4$, $b_3 = -1$, $b_4 = 3$, $b_5 = 4$, $b_6 = 1$ jne. Jos x esitetään kuitenkin muodossa

$$x = c_k 1000^k + c_{k-1}1000^{k-1} + ... + c_1 1000 + c_0,$$

missä $c_i \in \{0, 1, 2, ..., 999\}$ kaikilla $i = 0, 1, ..., k$, saadaan luvun x kolmellatoista jaollisuuden ehto nyt kaavojen (1) ja (2) nojalla muotoon

$$c_k(-1)^k + c_{k-1}(-1)^{k-1} + ... - c_1 + c_0 \equiv 0 \pmod{13}.$$

Jos x ja n ovat molemmat hyvin suuria lukuja, edellä kuvatusta tavasta konstruoida jaollisuustestejä ei ole suuresti apua, ellei lukua n voida esittää pienempien (alku-)lukujen tulona. Suuren luvun jakaminen tekijöihin on yleensä kuitenkin hyvin työlästä. On arvioitu, että tietokoneella, joka suorittaa miljardi operaatiota sekunnissa, 100-numeroisen luvun jakaminen tekijöihin kestää noin 26 vuorokautta ja 200-numeroisen luvun jakaminen vajaat 4 miljoonaa vuotta. Vielä vuonna 1988 200-numeroisen luvun tekijöihinjako olisi maksanut suurin piirtein saman verran kuin miehittämätön kuumatka.

Viitteet

1. P. Haukkanen: Algebran luentomoniste, Tampereen yliopisto, 2003.

2. V. Latvala ja P. Smolander: Modulaarisista laskutaulukoista, Solmu 2/2003.

3. J. Merikoski, A. Virtanen ja P. Koivisto: Diskreetti matematiikka I, Tampereen yliopisto, 1998.

4. J. Merikoski, K. Väänänen ja T. Laurinolli: Matematiikan taito 11, lukion pitkä matematiikka: Lukuteoria ja logiikka, Weilin+Göös, 3. p., 1997.

5. T. Metsäkylä: Kongruenssi – lukuteorian kätevä apuväline, Solmu 3/1997–1998.

6. K.H. Rosen: Discrete Mathematics and Its Applications, McGraw-Hill International Editions, 4th Ed., 1999.