UNIVERZA V MARIBORU
FAKULTETA ZA NARAVOSLOVJE IN MATEMATIKO
Oddelek za matematiko in računalništvo

Diplomsko delo

KOMBINATORIKA Z RAČUNALNIŠKIM PROGRAMOM MATHEMATICA

Mentorica:
izr. prof. dr. Petra Žigert

Kandidatka:
Dalija Jesenek

Maribor, 2009

ZAHVALA

Ne moremo vedno
delati velikih strvari,
lahko pa delamo
majhne stvari
z veliko ljubeznijo.

Mati Tereza

Iskreno se zahvaljujem dr. Petri Žigert za strokovno vodenje in pomoč pri nastajanju diplomskega dela.

Zahvala je namenjena tudi družini in fantu Roku za podporo in spodbudo v času šolanja.

UNIVERZA V MARIBORU
FAKULTETA ZA NARAVOSLOVJE IN MATEMATIKO

IZJAVA

Podpisana Dalija Jesenek, roj. 19.6.1985, študentka Fakultete za naravoslovje in matematiko Univerze v Mariboru, študijskega programa fizika in matematika, izjavljam, da je diplomsko delo z naslovom

KOMBINATORIKA Z RAČUNALNIŠKIM
PROGRAMOM MATHEMATICA

pri mentorici izr. prof. dr. Petri Žigert avtorsko delo. V diplomskem delu so uporabljeni viri in literatura korektno navedeni; teksti in druge oblike zapisov niso uporabljeni brez navedbe avtorjev.

Dalija Jesenek

Maribor, 2009

JESENEK, D.: Kombinatorika z računalniškim programom *Mathematica*. Diplomsko delo, Univerza v Mariboru, Fakulteta za naravoslovje in matematiko, Oddelek za matematiko in računalništvo, 2009.

IZVLEČEK

Diplomsko nalogo v grobem razdelimo na dva dela. Prvi del, ki obsega drugo in tretje poglavje, je namenjen spoznavanju programskega paketa *Mathematica*. V drugem poglavju opišemo zgradbo dokumenta (t.i. notebook) in palet, ki so namenjene lažjemu delu s programom. V nadaljevanju sledijo uporaba paketov, pravila sintakse in uvažanje ter izvažanje dokumentov. V tretjem poglavju obravnavamo računanje z osnovnimi matematičnimi operacijami in funkcijami vgrajenimi v paket. Predstavimo tudi temeljno računanje z izrazi, funkcijami, enačbami, matrikami, diferencialnim in integralnim računom. Zraven podamo ukaze in njihove razlage s konkretnimi primeri. V drugem delu (v četrtem in petem poglavju) podrobneje opišemo delo s paketom *Combinatorica* na področju kombinatorike in teorije grafov. V četrtem poglavju za lažjo obravnavo najprej predstavimo osnove kombinatorike ter ukaze povezane s kombinatoriko. V petem poglavju opišemo osnovne pojme teorije grafov in risanje grafov, katerih imena so že vgrajena v programu *Mathematica*. Nazadnje spoznamo risanje poljubnega grafa in ukaze za določanje Eulerjevega in Hamiltonovega grafa.

Ključne besede: program *Mathematica*, kombinatorika, grafi, diplomsko delo

JESENEK, D.: Combinatorics with the computer programme *Mathematica*.

Graduation Thesis, University of Maribor, Faculty of Natural Sciences and Mathematics, Department of mathematics and computer science, 2009.

ABSTRACT

This graduation thesis has been basically divided into two main parts, of which the First Part including the Chapters 2 and 3 has introduced the *Mathematica* package. The Chapter 2, however, introduces the construction of a document (i.e. the notebook) and pallets meant to less requiring way of dealing with the programme to be followed further by the use of the packages, the syntax regulations as well as by the importing and exporting documents. The Chapter 3 discusses package incorporated basic mathematical operations and functions involved in the calculating itself. Beside, fundamental calculating including expressions, functions, equations, matrixes as well as the differential and integral computation, which have been provided commands and explanations based on true examples have been subjected to the same chapter. The Second Part (Chapter 4 and 5), the operating with the *Combinatorica* package being involved into the field of the theory of combinations and the theory of graphs have been introduced. In order to make discussion easier, the Chapter 4 introduces the basis of the theory of combinations as well as related commands. The Chapter 5 clearly states basic conceptions on the theory of graphs and designing them already being incorporated into the *Mathematica*. Finally, the creation of an optional graph as well as the definition of the Euler and Hamilton graphs has been taught to sum up.

Key words: programme *Mathematica*, combinatorica, graphs, graduation thesis

Math. Subj.Class (2000): 05-04, 05A05, 05C45

PROGRAM DIPLOMSKE NALOGE

Diplomska naloga naj obravnava uvodne korake dela z računalniškim programom *Mathematica*. Jedro naloge naj predstavlja uporaba na področju kombinatorike.

Literatura:

S. Pemmaraju, S. Skiena, *Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*, Cambridge University Press, Cambridge, 2003

izr. prof. Petra Žigert

KAZALO

1. UVOD

Programski paket *Mathematica* je eden izmed mnogih programov (Derive, Matlab, ...) namenjenih za lažje delo na področju matematike. Razvil ge je Stephen Wolfram (rojen leta 1959). Prva različica *Mathematice* je bila izdana leta 1988 in se je na pričetku imenovala *Symbolic Manipulation Program* (*SMP*). Od tedaj je program doživel spremembo imena in kar nekaj posodobitev ter izboljšav.

Mathematica med drugim uporabniku omogoča simbolno in numerično računanje in izdelavo grafičnih predstavite (v 2D in 3D). Uporabimo jo lahko kot okolje za programiranje ali naložimo t.i. pakete, s katerimi razširimo možnosti uporabe za posamezno področje (npr. področje kombinatorike). Prav zato je uporaba *Mathematice* danes zelo razširjena. Največ se še vedno uporablja na znanstveno-raziskovalnem področju. Srečamo jo v naravoslovnih in tehničnih znanostih. Izjema pa niso niti družboslovne znanosti kot so ekonomija, sociologija in druge.

V izobraževalnih področjih je *Mathematica* dolgo veljala in še velja za prezahtevno, še posebno na srednješolskem nivoju. Mogoče tiči vzrok v učenju pravil pisanja matematičnih ukazov. Vendar se da osnove hitro naučiti. V pomoč so vedno priročne palete, s katerimi je možno zapisati veliko funkcij in ukazov. Navsezadnje pa ima *Mathematica* tudi zelo dobro pomoč (v obliki *Documentation Center*, *Function Navigator*, *Virtual Book* in *Find Selected Function*), kjer so navodila, razlage funkcij in ukazov podprte s konkretnimi primeri. Pričakovati, da se lahko uporabe *Mathematice* naučimo v celoti, je nerealno, saj je bistveno preobširna.

Na temo delati s programom *Mathematica* je napisanih že kar nekaj knjig. Le redke pa se poglobijo v področje kombinatorike. Prav zato bom v diplomski nalogi predstavila nekaj možnosti uporabe le te.

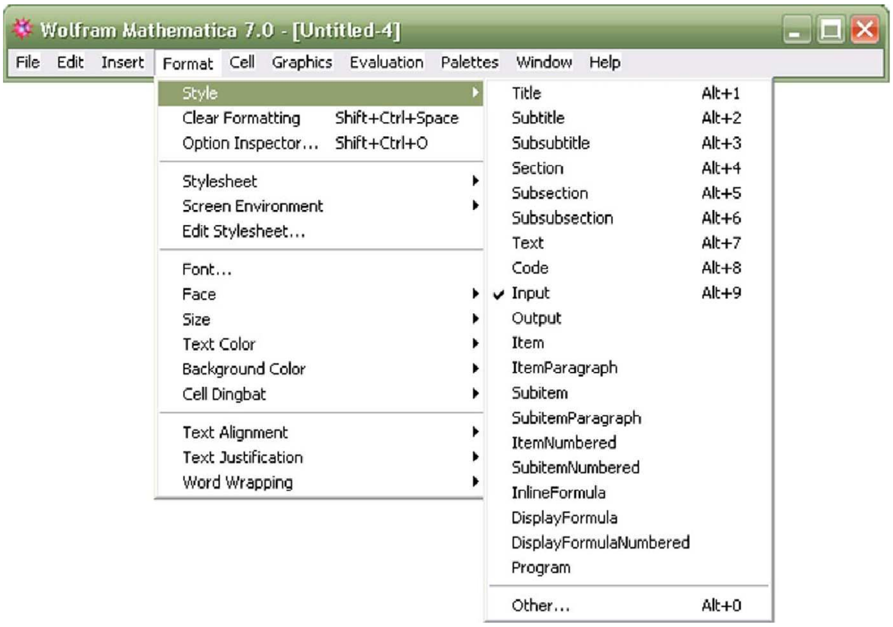
Najprej bomo spoznali zgradbo in osnove dela z *Mathematico*. V nadaljevanju pa računanje z izrazi, funkcijami, enačbami, matrikami, odvodi in integrali. Obravnavali bomo ukaze za štetje in zapisovanje urejenih in neurejenih izbir, kasneje pa še pomembnejše funkcije pri računanju s Fibonaccijevimi števili, Stirlingovimi števili prve in druge vrste ter permutacijami. Nato podamo ukaze za risanje grafov, ki so že vgrajeni v program *Mathematica*, in poljubnih grafov. Nazadnje obravnavamo ukaze za ugotavljanje Eulerjevega in Hamiltonovega grafa.

2. OSNOVE DELA Z MATHEMATICO

2.1. SESTAVA OKNA IN CELIC

Takoj po zagonu programa *Mathematica* se nam prikaže orodna vrstica z meniji *File, Edit, Insert*, itd. Navadno se nam samodejno odpre tudi prazen dokument (t.i. notebook oziroma zvezek). Ta ima končnico .nb, ki označuje datoteko *Mathematice*. V primeru, da se dokument sam ne odpre, ga najdemo pod *File-New-Notebook (.nb)*. Poleg notebooka nam *Mathematica* omogoča še izdelavo reprezentacije (*Slide Shov*) ali demonstracije (*Demonstration*). Prav tako ju najdemo pod *File-New*.

Tekst v dokumentu je razvrščen v celice, ki so označene z oglatimi zaklepaji ob desnem robu dokumenta. Poznamo več vrst celic. Največkrat uporabljene so tekstovne (*Text, Title, ...*), vhodne (*Input*) in izhodne (*Output*) celice, včasih pa še grafične, programske, kodne in druge. V vhodne celice pišemo ukaze, pri čemer se rezultati izpisujejo v izhodnih celicah. V tekstovne celice vnašamo spremno besedilo. Vrsto celice spremenimo tako, da označimo oklepaj celice in v meniju *Format-Style* (slika 1) izberemo željeno vrsto celice.



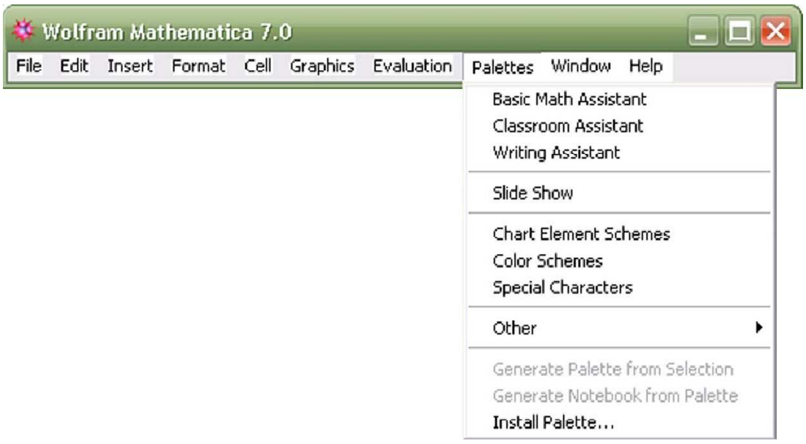
Slika 1: Sprememba vrste celice v meniju.

V obširnejših dokumentih je priporočljivo imeti besedilo razdeljeno na poglavja, podpoglavja in odstavke. To storimo z določitvijo celic kot so *Section*, *Subsection in Subsubsection*. Besedilo uredimo tudi z izbiro naslova (*Title*), podnaslova (*Subtitle*) in podpodnaslova (*Subsubtitle*). Vrsta celice, v katero pišemo navadno besedilo, je *Text*. Zgradba dokumenta je razvidna iz celic oziroma skupin celic na desnem robu okna (slika 2a). Celice in skupine celic so lahko odprte ali zaprte (slika 2b), glede na naše zanimanje. Med njimi lahko preklapljamo z dvojnim klikom na oglati oklepaj. Že napisane celice lahko združujemo ali razdružujemo ter vmes vstavljamo nove celice.

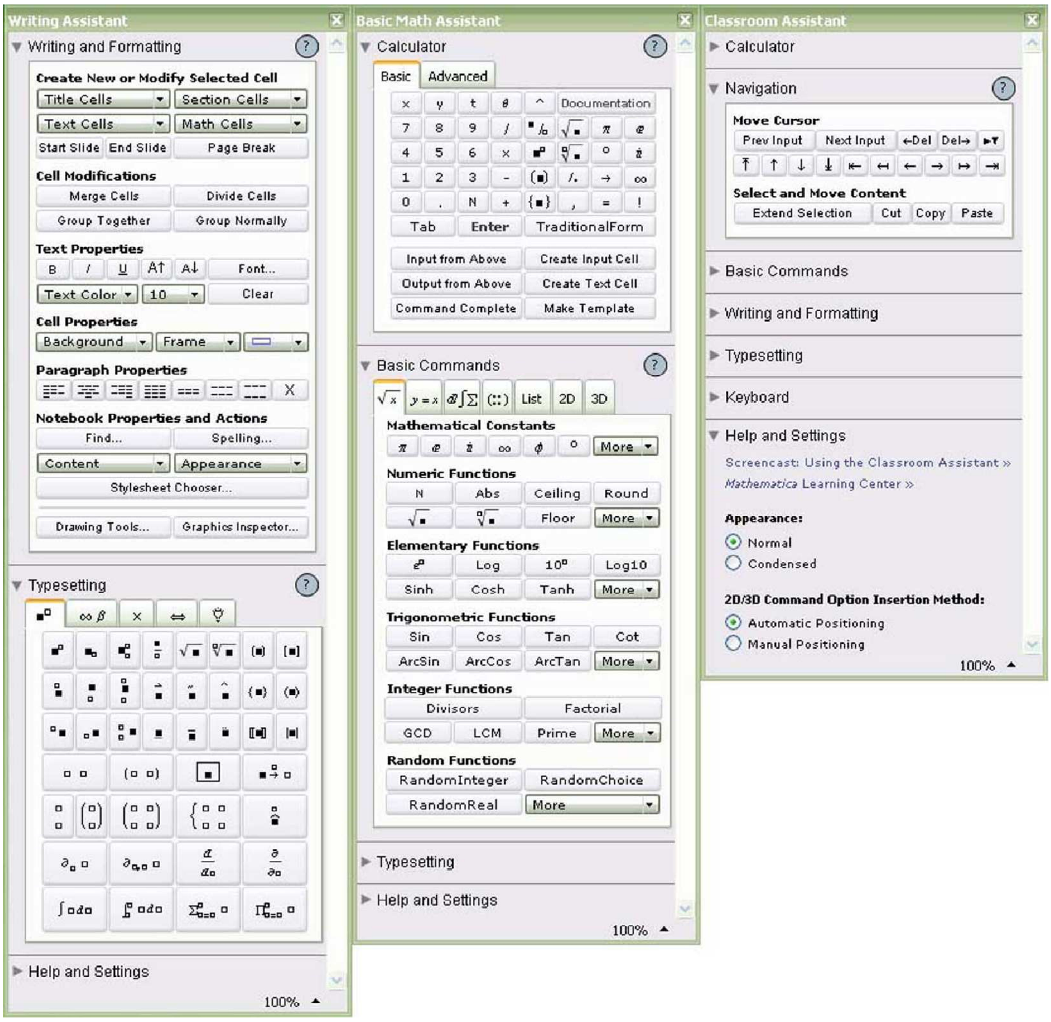
Slika 2: a) Vrste celic, b) zaprte celice.

2.2. PALETE

Delo s programskim paketom *Mathematica* si olajšamo z uporabo palet. V primerjavi s prejšnjo verzijo *Mathematico* 6 ima novejša verzija veliko več možnosti. Palete najdemo v meniju *Palettes* (slika 3). Najpomembnejši paleti za nas sta *Basic Math Assistant* in *Writting Assistant* (slika 4). Poleg njiju je na sliki tudi paleta *Classrom Assistant*, ki je mešanica obeh, prirejena za delo v razredu. Palete so razdeljene na več delov, ki jih po potrebi zapiramo in odpiramo. *Basic Math Assistant* je paleta z osnovnimi matematičnimi pripomočki. Vsebuje računalo (*Calculator*), osnovne matematične ukaze (*Basic Commands*) in pripomočke za lažje pisanje (*Typesetting*). V zadnjih najdemo predloge za različne matematične operacije kot so potenciranje, deljenje, korenjenje, odvajanje, integriranje, itd. Poleg tega pa pripomočki za lažje pisanje vsebujejo različne grške črke, matematične operatorje in puščice. Paleta *Writting Assistant* je namenjena urejevanju celic, oblikovanju besedila, preverjanju črkovanja ter osnovnemu risanju.



Slika 3: Izbira palet v meniju.



Slika 4: Palete *Basic Math Assistant*, *Writting Assistant* in *Classrom Assistant.*

2.3. PAKETI

Ena izmed najpomembnejših posebnosti *Mathematice* je, da je razširljiva. Standardna verzija *Mathematice* nam namreč zadostuje pri osnovnem računaju in uporabi funkcij. Če pa delamo v specializiranem območju matematike, kot je kombinatorika, lahko hitro ugotovimo, da vgrajene funkcije niso dovolj. Nove funkcije in ukaze moramo najprej naložiti s paketom (t.i. *Package*). To storimo z ukazom **Needs["Paket`"]**. Na sliki 5 so podani standardni paketi. Obstajajo še posebni, komercialni in uporabniški paketi, ki jih na tem seznamu paketov ni. Primer posebnega paketa je *MathLink*, ki omogoča povezavo med programom C in *Mathematico*. Komercialni paketi so plačniški in jih je potrebno dodatno kupiti (na internetu glej *The Wolfram Worldwide Web Store*). Uporabniške pakete pa uporabnik naredi sam.

Statistics Packages

- Analysis of Variance »
- Hierarchical Clustering »
- Hypothesis Testing »
- Multivariate Statistics »
- Statistical Plots »

Plotting, Charting, and Graphing Packages

- Error Bar Plotting »
- Plot Legends »
- Splines »

Discrete Math Packages

- *Combinatorica*
- Computational Geometry »
- Graph Utilities »

Calculus Packages

- Equation Trekker »
- Fourier Series »
- Function Approximations »
- Numerical Calculus »
- Numerical Differential Equation Analysis »
- Variational Methods »
- Vector Analysis »

Algebra Packages

- Finite Fields »
- Quaternions »

Polyhedra & Polytopes Packages

- Polyhedron Operations »
- Polytopes »

Miscellaneous Mathematical Packages

- Computer Arithmetic »
- Primality Proving »

Cartography & Dates Packages

- Calendar »
- Geodesy »
- World Plotting »

Sound Packages

- Audio »
- Music »

Physical Quantities & Properties Packages

- Black-Body Radiation »
- Physical Constants »
- Resonance Absorption Lines »
- Standard Atmosphere »
- Units »

Utility Packages

- Benchmarking »
- Developer Utilities »
- Experimental Functions »
- Notation »

Slika 5: Standardni paketi programa *Mathematica*.

2.4. PRAVILA PISANJA

Pred pričetkom pisanja opazimo, da ima miška v oknu vodoravno lego. To je znak, da lahko naredimo novo celico. Med celicami ima miška zopet vodoravno lego. Ko pričnemo pisati, se nam samodejno oblikuje vhodna celica. Izračun izraza sprožimo s pritiskom na tipko *Shift* + *Enter*, samo s tipko *Enter* pa skočimo v novo vrstico. Rešitev izraza se nam izpiše v izhodni celici. Vhodna in pripadajoča izhodna celica sta oštevilčeni z enako številko. V primeru, da kasneje med celice vstavimo novo vhodno celico, jo program oštevilči z zaporedno številko vpisa, pri čemer upošteva časovno zaporedje.

Kot drugi programi ima *Mathematica* pri pisanju v vhodne celice pravila sintakse. Ker loči med velikimi in malimi tiskanimi črkami, s slednjimi zapisujemo spremenljivke. Posledično konstante π, e, i označujemo z velikimi tiskanimi črkami (Pi, E, I). Funkcije vgrajene v *Mathematico* se začnejo z velikimi tiskanimi črkami (Sin, Intenger, Binomial, N, ...). Pri funkcijah sestavljenih iz več besed pričnemo z veliko začetnico vsako besedo (ArcTan, RandomSample, ...).

V vhodni celici ločimo dva načina zapisa ukaza s funkcijo. Prikazana sta v zgledu 2.1.

<u>Zgled 2.1:</u> Izračunaj sin (3π) na oba načina.

```
In[1]:= Sin[3 Pi]
Out[1]= 0
In[2]:= 3 Pi // Sin
Out[2]= 0
```

Bolj pregleden je prvi način, še posebno, ko delamo z več ukazi hkrati.

<u>Zgled 2.2:</u> Izračunaj določeni integral $\int_0^{3\pi}(\sin(x)+\cos(x))\,dx$.

```
In[3]:= Integrate[Sin[x] + Cos[x], {x, 0, 3 π}]
Out[3]= 2
```

Oklepaji imajo več nalog, ki so določene z njihovo obliko.

• ( ) oklepaje uporabljamo za razvrščanje v skupine,

• v [ ] oklepajih zapišemo argumente funkcij,

• v {} oklepajih naštevamo elemente skupine,

• v ⟦ ⟧ je podan indeks elementa skupine,

• v (**) podamo komentar.

<u>Zgled 2.3:</u> Prikaz uporabe različnih oklepajev.

```
In[4]:= ArcTan[1]  (*Izračunaj arctg(1).*)
Out[4]= π/4
```

```
In[5]:=  Permutations[{a, b}]  (*Zapiši permutacije
                                  elementov a in b.*)
Out[5]=  {{a, b}, {b, a}}
In[6]:=  {1, 5, 9}[[2]]      (*Iz skupine elementov 1,5 in 9
                               izberi število z indeksom 2.*)
Out[6]=  5
```

Z zavitimi oklepaji si lahko olajšamo večkratno računanje. Namesto da vstavljamo posamezne elemente, jih zapišemo kot seznam. Rezultati so zopet vrnjeni v obliki seznama.

Zgled 2.4: Seštej 1+ 3, 11.3 + 6,1 in $\frac{1}{4}+\frac{2}{3}$.

In[7]:= $\left\{1, 11.3, \frac{1}{4}\right\} + \left\{3, 6.1, \frac{2}{3}\right\}^2$

Out[7]= $\left\{10, 48.51, \frac{25}{36}\right\}$

Zgodi se, da potrebujemo rešitev naloge (enačbo, izraz, ...) večkrat. V novem računu nam ni potrebno pisati celotnega začetnega izraza znova. Namesto tega uporabimo oznako *%* kot nadomestilo za prejšnji rezultat, *%%* za predprejšnji rezultat itd. Če želimo uporabiti rezultat iz izhodne celice Out[n], kjer *n* označuje *n*-ti rezultat, krajše zapišemo *%n*.

Zgled 2.5: Prikaz uporabe %, %% in %*n*.

In[8]:= $\sqrt[4]{81} + 7$

Out[8]= 10

In[9]:= $\%^3$

Out[9]= 1000

In[10]:= $\frac{8 \cdot \%\%}{5}$

Out[10]= $\frac{8 \cdot 10}{5}$

In[11]:= $e^{\%9}$

Out[11]= e^{1000}

Med drugim lahko rezultat (enačbo, izraz, ...) tudi poimenujemo in v nadaljevanju namesto ponovnega pisanja celotnega začetnega izraza (enačbe, ...) uporabljamo le ime.

Zgled 2.6: Izračunaj izraz log $(a+9)$, kjer je a rezultat izraza $\sqrt[3]{27}+7$.

In[12]:= **ime = $\sqrt[3]{27}$ + 7**

Out[12]= 10

In[13]:= **Log[ime + 9]**

Out[13]= Log[19]

Opis funkcije ali simbola dobimo z ukazom **?ime**, podrobnejši opis pa z **??ime**. V primeru, da poznamo le začetek imena funkcije uporabimo **?im***.

Zgled 2.7: a) Osnovni in b) podrobnejši opis limite ter c) seznam vseh funkcij, ki se pričnejo s črkami Lim.

In[14]:= **? Limit**

Limit[*expr*, *x* –> x_0] finds the limiting value of *expr* when *x* approaches x_0. >>

In[15]:= **?? Limit**

Limit[*expr*, *x* –> x_0] finds the limiting value of *expr* when *x* approaches x_0. >>

```
Attributes[Limit] = {Listable, Protected}

Options[Limit] =
 {Analytic → False, Assumptions :→ $Assumptions, Direction → Automatic}
```

In[16]:= **? Lim***

▼ System`

Limit	LimitsPositioning	LimitsPositioningTokens

2.5. UVOZ IN IZVOZ DOKUMENTOV

Mathematica omogoča izvoz posameznih izrazov v druge programe (na primer *Latex*, *Mathlink*) in uvoz izrazov iz drugih programov v *Mathematico*. Pri izvozu izraz najprej pretvorimo npr. v *Latex* obliko z ukazom **TeXForm[izraz]** in šele nato kopiramo v *Latex* dokument. Pri uvozu pa izraz najprej kopiramo v *Mathematico* in nato pretvorimo v ustrezno obliko *Mathematice* z ukazom **ToExpression["\\izraz", TeXForm]**.

Zgled 2.8: Izraz $\sqrt{3+x} + \frac{(1+x)^2}{2x}$ v *Mathematici* pretvorimo v *Latex* obliko

In[17]:= $\texttt{Sqrt[3 + x] +} \dfrac{\texttt{(x + 1)^2}}{\texttt{2 x}}$

Out[17]= $\dfrac{(1+x)^2}{2\,x} + \sqrt{3+x}$

In[18]:= **TeXForm[%]**

Out[18]//TeXForm=

```
\frac{(x+1)^2}{2 x}+\sqrt{x+3}
```

in obratno.

In[19]:=
```
ToExpression[
 "\\frac{(x+1)^2}{2 x}+\sqrt{x+3}", TeXForm]
```

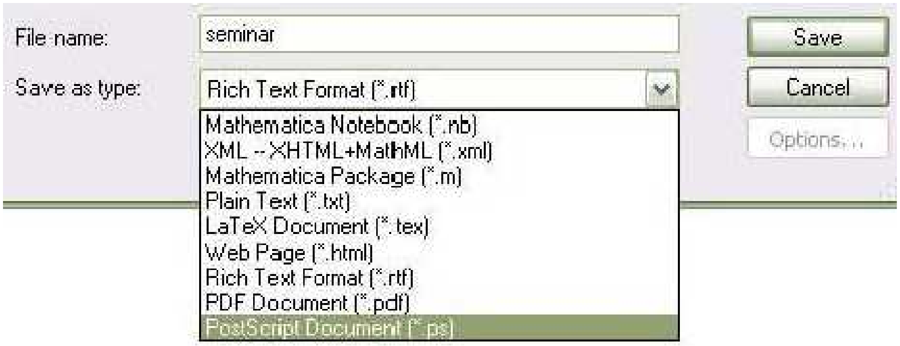
Out[19]= $\dfrac{(1+x)^2}{2\,x} + \sqrt{3+x}$

V tabeli so predstavljeni ukazi za pretvorbo izrazov v obliko nekaterih programov:

Simbol	Oblike
TraditionalForm[izraz]	tradicionalna matematična oblika
TeXForm[izraz]	TEX oblika
MathMLForm[izraz]	MathML oblika
CForm[izraz]	C oblika
FortranForm[izraz]	Fortran oblika

Včasih je bolje, da za povezavo med *Mathematico* in programom raje uporabimo kar t.i. paket *MathLink*. To lahko storimo tedaj, ko imamo program, ki je združljiv z *MathLink*-om.

Mathematica je zmožna pretvoriti tudi celoten dokument in ga shraniti kot dokument drugega programa. To lahko navadno storimo z izbiro *File→Save As* (slika 6). Pri pretvorbi dokumenta napisanega v slovenskem jeziku, ima lahko program probleme. Problematične so še posebno črke š, č in ž.



Slika 6: Tipi dokumentov v katere lahko shranimo dokument *Mathematice.*

Celoten zvezek lahko shranimo v drugo obliko tudi z ukazom v vhodni celici. To storimo z **Export["ime.pdf", EvaluationNotebook[]]**. Pri tem se nam ukaz v shranjenem dokumentu prav tako izpiše.

Zgled 2.9: Zvezek shranimo v pdf dokument z imenom Diploma.

```
In[20]:= Export["Diploma.pdf", EvaluationNotebook[]]
Out[20]= Diploma.pdf
```

3. ALGEBRA IN ANALIZA Z MATHEMATICO

3.1. OSNOVNE OPERACIJE IN FUNKCIJE

Preprosto računanje z *Mathematico* je podobno kot s kalkulatorjem. Za operacije množenja, deljenja, seštevanja, odštevanja in potenciranja uporabljamo simbole *, /, +, -, ^. Simbol * lahko nadomestimo s presledkom. Operacije deljenja, potenciranja in korenjenja lahko zapišemo tudi s pomočjo palet. Glej spodnjo tabelo in zgled 3.1:

OPERACIJA	S SIMBOLOM	S PALETO
deljenje	x / y	$\frac{x}{y}$
potenciranje	x^y	x^y
korenjenje	Sqrt[x] x^(1 / y)	$\sqrt{x}$ $\sqrt[y]{x}$

<u>Zgled 3.1:</u>

In[21]:= $\frac{25}{3}$

Out[21]= $\frac{25}{3}$

In[22]:= 5^{40}

Out[22]= 9 094 947 017 729 282 379 150 390 625

In[23]:= **Sqrt[Pi]**

Out[23]= $\sqrt{\pi}$

In[24]:= $\sqrt[3]{7}$

Out[24]= $7^{1/3}$

In[25]:= **23.43 / 7**

Out[25]= 3.34714

Opazimo, da *Mathematica* zapiše razultat kot natančno vrednost (brez decimalnih mest). Izjema je le, če deljenec ali delitelj nastopa kot decimalno število. Tedaj *Mathematica* rezultat v izhodni celici zaokroži na število decimalnih mest, ki je določeno v lastnostih *Mathematice*. Če želimo, da se natančne vrednosti izpišejo z decimalnimi števili, uporabimo ukaz **//N** ali **N[izraz]**. Število *n* željenih decimalnih mest določimo z ukazom **N[izraz, n]**.

Zgled 3.2: Zapiši

a) števili $\frac{25}{3}$ in 5^{40} kot decimalni števili,

b) število π kot decimalno število na deset mest natančno.

In[26]:= $\frac{25}{3}$ // N

Out[26]= 8.33333

In[27]:= N[5^{40}]

Out[27]= 9.09495×10^{27}

In[28]:= N[Pi, 10]

Out[28]= 3.141592654

Rezultati se nam včasih izpišejo v več vrsticah. **Short[izraz]** nam poda le začetne in končne števke rezultata. Med oklepaji **≪ ≫** zapiše, koliko števk je izpustil.

Zgled 3.3: Izračunaj 8^{300}. Rezultat zapiši še v krajši obliki.

In[29]:= 8^{300}

Out[29]= 8 452 712 498 170 643 941 637 436 558 664 265 704 301 557 216 \
577 944 354 047 371 344 426 782 440 907 597 751 590 676 094 \
202 515 006 314 790 319 892 114 058 862 117 560 952 042 968 \
596 008 623 655 407 033 230 534 186 943 984 081 346 699 704 \
282 822 823 056 848 387 726 531 379 014 466 368 452 684 024 \
987 821 414 350 380 272 583 623 832 617 294 363 807 973 376

In[30]:= Short[8^{300}]

Out[30]//Short=
84 527 124 981 706 439 416 374 365 586 642 657 043 015 572 165 \
779 443 540 473 713 444 ≪140≫
379 014 466 368 452 684 024 987 821 414 350 380 272 583 623 \
832 617 294 363 807 973 376

3.2. FUNKCIJE

Funkcije uporabljamo na mnogih področjih matematike in na zelo različne načine. V programu *Mathematica* imamo nekatere funkcije že vgrajene v program, vedno pa lahko definiramo nove. Da jih ločimo od prvih, zanje uporabljamo male tiskane črke.

Novo funkcijo najprej definiramo. Pri tem zapišemo argument v oglati oklepaj. Znotraj oklepajev ima argument na desni strani še podčrtaj. Na desni strani enakosti nastopa spremenljivka brez podčrtaja.

Zgled 3.4: Izračunj vrednost funkcije $f(x) = x^2$ za $x = 5$.

In[31]:= **f[x_] = x^2**

Out[31]= x^2

In[32]:= **f[5]**

Out[32]= 25

Pred ponovnim definiranjem funkcije z istim imenom moramo izbrisati prejšnjo.

Zgled 3.5: Izračunj vrednost funkcije $f(x) = x^3 + 2$ za števila 7, π in $y + 1$.

In[33]:= **Clear[f]**

In[34]:= **f[x_] = x^3 + 2**

Out[34]= 2 + x^3

In[35]:= **{f[7], f[Pi], f[y + 1]}**

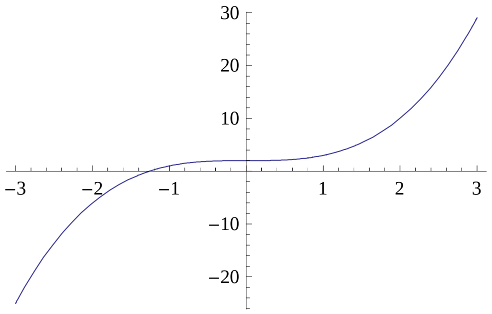
Out[35]= $\{345,\ 2 + \pi^3,\ 2 + (1 + y)^3\}$

Funkcijo f spremenljivke x med vrednostima $x_{\min}$ in $x_{\max}$ narišemo z ukazom **Plot[f,{x, x_{min}, x_{max}}]**, več funkcij iste spremenljivke pa narišemo s **Plot[{ f_1, f_2, ... }**,**{x, x_{min}, x_{max}}]**.

Zgled 3.6: Nariši funkcijo $f(x) = x^3 + 2$ na intervalu [-3, 3].

In[36]:= **Plot[x^3 + 2, {x, -3, 3}]**

Out[36]=



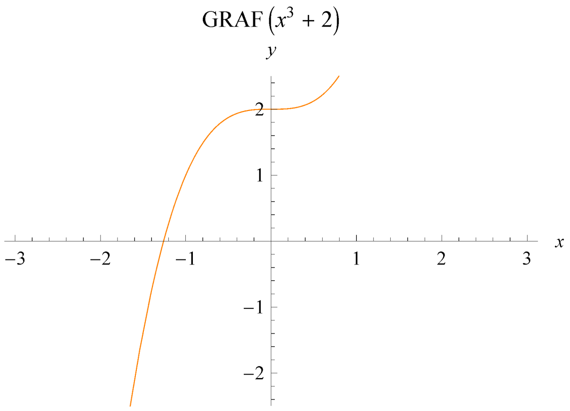
Pri risanju grafov imamo mnogo dodatnih možnosti. V tabeli so zapisane le nekatere.

Ukaz	**Opis ukaza**
Axes → False	nariše graf brez osi
Axes → {True, False}	x os nariše, y os ne nariše
AxesLabel → {x, y}	na grafu označi osi z x in y
PlotLabel → Moj graf	izpiše naslov grafa : Moj graf
AxesOrigin → {1, 2}	določi točko kjer se sekajo osi : točka {1, 2}
AxesStyle → {Red, Blue}	določi barvo osi : x os je rdeča, y os je modra
ColorFunction → Function[{x, y}, Orange]	določi barvo funkcije v ravnini : oranžna
PlotRange → Automatic	prilagodi velikost osi grafu
PlotRange → 2	izriše graf med $y = -2$ in $y = 2$
Frame → True	okoli grafa izriše okvir

<u>Zgled 3.7:</u> Nariši funkcijo $f(x) = x^3 + 2$ na intervalu [-3, 3]. Narisana naj bo z oranžno barvo, z označenima osema, naslov grafa naj bo GRAF $x^3 + 2$. Graf izriši med y = - 2,5 in y = 2,5.

In[37]:=
```
Plot[x^3 + 2, {x, -3, 3},
  ColorFunction → Function[{x, y}, Orange],
  AxesLabel → {x, y},
  PlotLabel → GRAF (x^3 + 2), PlotRange → 2.5]
```

Out[37]=



Tvorimo lahko tudi funkcije z več spremenljivkami.

Zgled 3.8: Izračunj vrednost funkcije $f(x) = x^2 + y^2$ za točko (2, 4i).

In[38]:= `f[x_, y_] = x^2 + y^2`

Out[38]= $x^2 + y^2$

In[39]:= `f[2, 4 I]`

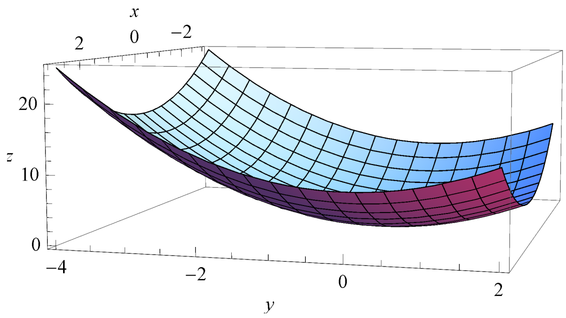
Out[39]= -12

Funkcijo f spremenljivk x in y v prostoru narišemo podobno kot funkcijo ene spremenljivke, in sicer s **Plot3D[f, {x,** x_{min}**,** x_{max}**}, {y,** y_{min}**,** y_{max}**}]**. Pri tem moramo podati meje za obe spremenljivki. S **Plot3D[{** f_1**,** f_2**, ... }, {x,** x_{min}**,** x_{max}**}, {y,** y_{min}**,** y_{max}**}]** narišemo več funkcij istih spremenljivk.

Zgled 3.9: Nariši funkcijo $f(x) = x^2 + y^2$ med $-3 \le x \le 3$ in med $-4 \le y \le 2$.

In[40]:= `Plot3D[x^2 + y^2, {x, -3, 3}, {y, -4, 2}, AxesLabel → {x, y, z}]`

Out[40]=



3.3 IZRAZI

V *Mathematici* poznamo tako simbolično kot numerično računanje izrazov. Simbolno seštevanje je prikazano z naslednjim zgledom.

Zgled 3.10: Poenostavi izraz $3 + 5a - 7b + c - \frac{24a^2}{8a} + \sqrt{81}\, b$.

In[41]:= $3 + 5a - 7b + c - \frac{24a^2}{8a} + \sqrt{81}\, b$

Out[41]= $3 + 2a + 2b + c$

Simbole v izrazu lahko nadomestimo z drugo vrednostjo. To storimo z ukazom **Replace[izraz, a -> vrednost]** ali **izraz /. a->vrednost**, kjer je *a* zamenjani simbol. Več simbolov hkrati zamenjamo z **izraz /. {a -> avrednost, b -> bvrednost, ...}**.

Zgled 3.11: a) V izrazu a^{2+b} zamenjaj a^{2+b} z $b+3$.
b) V izrazu $3 + 2a + 2b + c$ zamenjaj a z 6, b z $\frac{1}{2}$ in c z $0.4+r$.

```
In[42]:= Replace[a^(2+b), a^(2+b) → b + 3]
Out[42]= 3 + b
In[43]:= 3 + 2 a + 2 b + c /. {a -> 6, b -> 1 / 2, c -> 0.4 + r}
Out[43]= 16.4 + r
```

Ukaz Replace nam ne nadomesti dela izraza, kot je prikazano v zgledu 3.12. Vrne nam prvotni izraz.

Zgled 3.12: V izrazu nadomesti člen x^3 z 5 + a.

```
In[44]:= Replace[6 + x^3, x^3 -> 5 + a]
Out[44]= 6 + x^3
```

V argument je potrebno dodati še **{1}**, kar pomeni zamenjavo na prvi stopnji.

Zgled 3.13: V izrazu nadomesti člen x^3 z 5 + a.

```
In[45]:= Replace[6 + x^3, x^3 -> 5 + a, {1}]
Out[45]= 11 + a
```

Osnovni ukazi za računanje z izrazi so še **Expand[izraz]**, **Factor[izraz]** in **Simplify[izraz]**. Naloga prvega je, da pomnoži vse produkte in izraze v oklepajih ter jih zapiše kot vsoto členov. Drugi ukaz zapiše izraz kot produkt najmanjših faktorjev. Ukaz Simplify pa skrči izraz v najpreprostejšo obliko. Pri tem uporablja standardne algebraične transformacije kot so faktorizacija, krajšanje in razširjanje ulomkov, iskanje skupnega imenovalca, izpostavljanje in še mnoge druge. Te ukaze lahko uporabljamo tudi ločeno. V primeru, da nam ukaz Simplify ne poenostavi dovolj rezultata, uporabimo ukaz **FullSimplify[izraz]**. Ukazi s Simplify so med najbolj uporabnimi v *Mathematici*.

Zgled 3.14: Izraz $\left(a^2-1\right)^2 \cdot \left(1+a^2+a^4\right)^2$ prvič pomnoži, drugič razstavi in tretjič poenostavi.

In[46]:= `Expand[(a^2 - 1)^2 (1 + a^2 + a^4)^2]`

Out[46]= $1 - 2\,a^6 + a^{12}$

In[47]:= `Factor[%]`

Out[47]= $(-1+a)^2\,(1+a)^2\,\left(1-a+a^2\right)^2\,\left(1+a+a^2\right)^2$

In[48]:= `Simplify[%%]`

Out[48]= $\left(-1+a^6\right)^2$

3.4. ENAČBE

Pri simbolnem reševanju enačb uporabljamo ukaz **Solve[enačba, {x, y, ...}]**, kjer so {*x*, *y*, ...} neznanke enačbe. Če je neznanka ena sama, je ni potrebno pisati v zavitem oklapaju. Enačbe zapišemo v obliki **Leva stran enačbe==Desna stran enačbe**, pri čemer za enakost uporabimo dvojni enačaj. Enojni enačaj namreč uporabljamo, ko prirejamo enačbam, izrazom, funkcijam itd. imena.

Zgled 3.15: Zapis enačbe $(x+3)^2 = 13+(2\,x-4)^2$ v *Mathematici*.

In[49]:= `(x + 3)^2 == 13 + (2 x - 4)^2`

Out[49]= $(3+x)^2 == 13 + (-4+2\,x)^2$

Ko rešujemo sistem enačb, enačbe zapišemo v obliki seznama. V njem so enačbe med seboj ločene z vejicami. Za reševanje sistema enačb uporabimo ukaz **Solve[{enačba1, enačba2, ...}, {x, y, ...}]**. Rešitve sistema se prav tako zapišejo v obliki seznama.

Zgled 3.16: Reši

a) enačbo $(x+3)^2 = 13+(2\,x-4)^2$,
b) sistem dveh enačb $2\,x - 3\,y + 3 = 0$ ter $2\,y+5\,x=3$.

In[50]:= `Solve[(x + 3)^2 == 13 + (2 x - 4)^2, x]`

Out[50]= $\left\{\left\{x \to \frac{1}{3}\left(11-\sqrt{61}\right)\right\}, \left\{x \to \frac{1}{3}\left(11+\sqrt{61}\right)\right\}\right\}$

```
In[51]:= Solve[{2 x - 3 y + 3 == 0, 2 y + 5 x == 3}, {x, y}]
```

Out[51]= $\left\{\left\{x \to \frac{3}{19},\ y \to \frac{21}{19}\right\}\right\}$

Včasih nam zgornji ukaz ne izpiše vseh rešitev, zato za vse možne rešitve uporabimo ukaz **Reduce[{enačba1, enačba2, ...}, {x, y, ...}]**, kjer so {*x*, *y*, ...} vse neznanke. Ukaz **Eliminate[{enačba1, enačba2, ...}, {x, y, ...}]** pa nam sistem enačb skrči in neznanke {*x*, *y*, ...} eliminira.

Zgled 3.17: a) Reši sistem enačb $x^2 - b = a$ in $3\,x = b^2$.
b) V sistemu enačb $6\,x + 4\,y - 3\,z = 0$, $-18\,x + 5\,y - 3\,z = 2$ in $x + 7\,y = 0$ eliminiraj spremenljivko x.

In[52]:= **Solve**$\left[\left\{x^2 - b == a,\ 3\,x == b^2\right\},\ x\right]$

Out[52]= { }

In[53]:= **Reduce**$\left[\left\{x^2 - b == a,\ 3\,x == b^2\right\},\ x\right]$

Out[53]= $a == \frac{1}{9}\left(-9\,b + b^4\right)\ \&\&\ x == \frac{b^2}{3}$

```
In[54]:= Eliminate[
  {6 x + 4 y - 3 z == 0, -18 x + 5 y - 3 z == 2, x + 7 y == 0}, x]
```

Out[54]= $38\,y == -3\,z\ \&\&\ 507\,z == -76$

V zgornjih primerih znak && pomeni in. Znak || pa bi pomenil ali.

Enačbe in sistemi enačb, ki vsebujejo polinome stopnje štiri ali manj, so vedno rešljivi z ukazom Solve. Če rešujemo polinome stopnje pet ali več, *Mathematica* ne more podati točne formule za vse rešitve. Tedaj enačbe ali sistem enačb rešimo numerično z uporabo ukaza **NSolve[enačba, {x, y, ...}]** oziroma **NSolve [{enačba1,enačba2, ...}, {x, y, ...}]**. V ukaz lahko dodamo parameter *n*, s katerim določimo število mest pri zaokroževanju rezultata.

Zgled 3.18: Numerično reši

a) enačbo $x^5 + 7x^3 + 4x^2 + 6 = x$,

b) sistem enačb $4x^3 + 3y^2 = 3$ ter $x^2 - y^2 = 2$. Rezultat zaokroži na tri mesta.

```
In[55]:= NSolve[x^5 + 7 x^3 + 4 x^2 + 6 == x, x]

Out[55]= {{x → -1.14164}, {x → 0.215616 - 2.6885 i},
          {x → 0.215616 + 2.6885 i},
          {x → 0.355206 - 0.7722 i}, {x → 0.355206 + 0.7722 i}}

In[56]:= NSolve[{4 x^3 + 3 y^2 == 3, x^2 - y^2 == 2}, {x, y}, 3]

Out[56]= {{x → -0.93 - 1.09 i, y → 0.61 + 1.64 i},
          {x → -0.93 + 1.09 i, y → 0.61 - 1.64 i},
          {x → 1.10, y → 0. × 10^-4 + 0.89 i},
          {x → 1.10, y → 0. × 10^-4 - 0.89 i},
          {x → -0.93 + 1.09 i, y → -0.61 + 1.64 i},
          {x → -0.93 - 1.09 i, y → -0.61 - 1.64 i}}
```

Enačbe in sisteme enačb je mogoče preoblikovati v implicitno obliko in poiskati njihove ničle. Ničle enačbe najdemo z ukazom **FindRoot [enačba, {x, x_0}]** ali **FindRoot [enačba, {x, {x_0, y_0} }]**. Ukaza se razlikujeta v številu začetnih pogojev. Prvi išče rešitve v okolici vrednosti x_0 z uporabo Newtonove metode. Drugi ukaz pa išče ničlo med vrednostima x_0 in y_0, in sicer z uporabo sekantne metode. Podobno poiščemo ničle sistemov enačb.

Zgled 3.19: a) Poišči ničle enačbe $(x+3)^2 = 13 + (2x-4)^2$ v okolici 1 in na intervalu [1, 10].

b) Poišči ničlo sistema enačb $3x + y - z = 1$, $x - y - 5z = -1$, $6x + 7y + 9z = 3$ za x v okolici števila 1, za y v okolici števila 3 in za z v okolici števila 2.

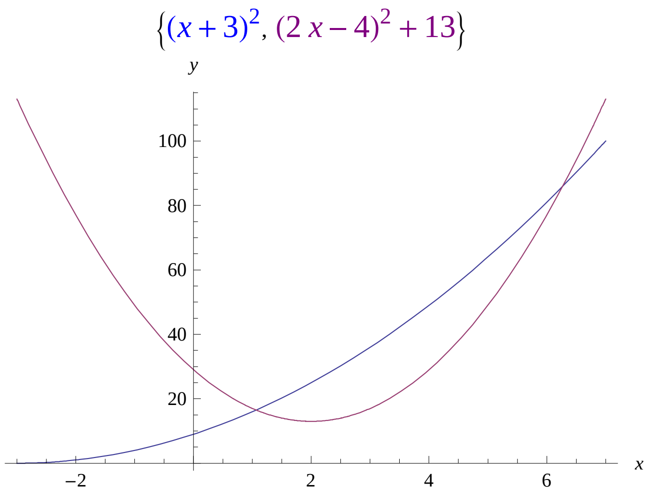
```
In[57]:= FindRoot[(x + 3)^2 == 13 + (2 x - 4)^2, {x, 1}]

Out[57]= {x → 1.06325}

In[58]:= FindRoot[(x + 3)^2 == 13 + (2 x - 4)^2, {x, {1, 10}}]

Out[58]= {x → {1.06325, 6.27008}}

In[59]:= FindRoot[{3 x + y - z == 1, x - y - 5 z == -1, 6 x + 7 y + 9 z == 3},
          {{x, 1}, {y, 3}, {z, 2}}]

Out[59]= {x → 0.923077, y → -1.15385, z → 0.615385}
```

Rešitev enačbe pa lahko poiščemo tudi grafično z ukazom **Plot[{leva stran enačbe, desna stran enačbe},{x, x$_{\mathbf{min}}$, x$_{\mathbf{max}}$}]**. Z $x_{\min}$ in $x_{\max}$ določimo interval na abcisni osi, v katerem nam *Mathematica* nariše grafa za levo in desno stran enačbe. Abcisi točk, kjer se grafa za levo in desno stran enačbe sekata, sta rešitvi enačbe.

<u>Zgled 3.20:</u> Grafično poišči rešitev enačbe $(x + 3)^2 = 13 + (2\,x - 4)^2$.

In[60]:=
```
Plot[{(x + 3)^2, 13 + (2 x - 4)^2},
 {x, -3, 7}, AxesLabel → {x, y},
 PlotLabel → {Style[(x + 3)^2, Blue, 15],
   Style[13 + (2 x - 4)^2, Purple, 15]}]
```

Out[60]=



3.5. MATRIKE

V *Mathematici* matrike zapišemo na več načinov. Prva možnost je zapis matrike *a* v obliki seznama po vrsticah, ki jih nato z ukazom **MatrixForm[a]** zapišemo v nam znano obliko. Oba koraka lahko združimo v enega (zgled 3.21.).

<u>Zgled 3.21:</u>

In[61]:=
```
a = {{2, -6, 7}, {0, 1, 8}}
```

Out[61]= {{2, -6, 7}, {0, 1, 8}}

In[62]:=
```
MatrixForm[a]
```

Out[62]//MatrixForm=

$$\begin{pmatrix} 2 & -6 & 7 \\ 0 & 1 & 8 \end{pmatrix}$$

Druga možnost je z vstavljanjem matrike iz palete ali menija (*Insert*→*Table*/*Matrix*→ *New*). Po potrebi dodamo nove stolpce in vrstice (*Insert*→*Table*/*Matrix*→*Add Row* ali *Add Column*).

Posebne ukaze imamo za zapis diagonalnih in enotskih matrik ter matrik, kjer so vsi elementi enaki. Glej zgled 3.22.

<u>Zgled 3.22:</u> Zapiši

a) diagonalno matriko z diagonalnimi elementi 1, 2 in 3,

b) enotsko matriko reda 4×4 in

c) matriko reda 3×2, pri kateri so vsi elementi enaki *r*.

```
In[63]:= DiagonalMatrix[{1, 2, 3}] // MatrixForm
```

Out[63]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$$

```
In[64]:= IdentityMatrix[4] // MatrixForm
```

Out[64]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
In[65]:= ConstantArray[r, {3, 2}] // MatrixForm
```

Out[65]//MatrixForm=

$$\begin{pmatrix} r & r \\ r & r \\ r & r \end{pmatrix}$$

Matriki *a* zamenjamo stolpce in vrstice oziroma jo transponiramo z ukazom **Transpose[a]**.

<u>Zgled 3.23:</u> Transponiraj matriko $B = \begin{pmatrix} 1 & -2 & 4 \\ 3 & 1 & 3 \end{pmatrix}$.

```
In[66]:= b = {{1, -2, 4}, {3, 1, 3}}
Out[66]= {{1, -2, 4}, {3, 1, 3}}
```

In[67]:= `MatrixForm[b]`

Out[67]//MatrixForm=

$$\begin{pmatrix} 1 & -2 & 4 \\ 3 & 1 & 3 \end{pmatrix}$$

In[68]:= `Transpose[b] // MatrixForm`

Out[68]//MatrixForm=

$$\begin{pmatrix} 1 & 3 \\ -2 & 1 \\ 4 & 3 \end{pmatrix}$$

Osnovne operacije matrik so seštevanje, odštevanje in množenje matrik. Pri seštevanju in odštevanju sta operatorja plus in minus, medtem ko za množenje uporabimo kar piko.

Zgled 3.24: Izračunaj $A+B \cdot F^T$, kjer je $A = \begin{pmatrix} 0 & 2 \\ -3 & 5 \\ 1 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 2 & -5 & -1 \\ 3 & 2 & -2 & 0 \\ 1 & -1 & 0 & 4 \end{pmatrix}$ in $C = \begin{pmatrix} 0 & -1 & 4 & 5 \\ 2 & -1 & 2 & 2 \end{pmatrix}$.

In[69]:= `Clear[a, b]`

In[70]:= $a = \begin{pmatrix} 0 & 2 \\ -3 & 5 \\ 1 & 1 \end{pmatrix}$; $b = \begin{pmatrix} 1 & 2 & -1 & -1 \\ 3 & 2 & -2 & 0 \\ 1 & -1 & 0 & 4 \end{pmatrix}$; $c = \begin{pmatrix} 0 & -1 & 4 & 1 \\ 2 & -1 & 2 & 2 \end{pmatrix}$

`2 a - 5 b.Transpose[c] // MatrixForm`

Out[70]= `{{0, -1, 4, 1}, {2, -1, 2, 2}}`

Out[71]//MatrixForm=

$$\begin{pmatrix} 35 & 24 \\ 44 & 10 \\ -23 & -53 \end{pmatrix}$$

Ukaz za inverzno matriko matrike *a* je **Inverse[a]**, ukaz za izračun determinante je **Det[a]**, vsoto elementov po glavni diagonali pa nam poda sled matrike **Tr[a]**.

Zgled 3.25: Preveri ali ima matrika $D = \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 5 \\ 1 & 3 & -2 \end{pmatrix}$ inverz ter ga izračunaj. Izračunaj tudi sled matrike.

```
In[72]:= d = {{1, -3, 2}, {0, 2, 5}, {1, 3, -2}}

Out[72]= {{1, -3, 2}, {0, 2, 5}, {1, 3, -2}}

In[73]:= Det[d] // MatrixForm

Out[73]//MatrixForm=
         -38
```

Determinanta je različna od nič, zato ima matrika D inverz.

```
In[74]:= Inverse[d] // MatrixForm

Out[74]//MatrixForm=
```

$$\begin{pmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ -\frac{5}{38} & \frac{2}{19} & \frac{5}{38} \\ \frac{1}{19} & \frac{3}{19} & -\frac{1}{19} \end{pmatrix}$$

```
In[75]:= Tr[d]

Out[75]= 1
```

Največje število linearno neodvisnih stolpcev oziroma vrstic (t.i. rang matrike) matrike *a* poiščemo z ukazom **MatrixRank[a]**.

Zgled 3.26: Izračunaj rang matrike $E = \begin{pmatrix} 3 & -1 & 7 & 2 \\ 4 & 0 & 1 & -8 \end{pmatrix}$.

```
In[76]:= e = {{3, -1, 7, 2}, {4, 0, 1, -8}}
         MatrixRank[e]

Out[76]= {{3, -1, 7, 2}, {4, 0, 1, -8}}

Out[77]= 2
```

Matriko *a* preuredimo v zgornjetrikotno matriko s **UpperTriangularize[a],** medtem ko eliminiramo število vrstic z **RowReduce[a]**.

Zgled 3.27: Preuredi matriko $F = \begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 5 \\ -1 & 3 & -2 \end{pmatrix}$ v zgornjetrikotno matriko, zatem matriki F eliminiraj vrstice.

```
In[78]:= f = {{1, -3, 2}, {0, 2, 5}, {-1, 3, -2}}
Out[78]= {{1, -3, 2}, {0, 2, 5}, {-1, 3, -2}}
In[79]:= UpperTriangularize[f] // MatrixForm
Out[79]//MatrixForm=
```

$$\begin{pmatrix} 1 & -3 & 2 \\ 0 & 2 & 5 \\ 0 & 0 & -2 \end{pmatrix}$$

```
In[80]:= RowReduce[f] // MatrixForm
Out[80]//MatrixForm=
```

$$\begin{pmatrix} 1 & 0 & \frac{19}{2} \\ 0 & 1 & \frac{5}{2} \\ 0 & 0 & 0 \end{pmatrix}$$

```
In[81]:= Clear[a, b, c, d, e, f]
```

3.6. DIFERENCIALNI RAČUN

f '(x) je eden izmed zapisov odvoda funkcije spremenljivke *x*. V Mathematici funkcije ene spremenljivke odvajamo z ukazi **f '[x]** = **D[f[x], x]** = ∂_x**f[x]**. Zadnja ukaza sta ukaza za parcialno odvajanje.

Zgled 3.28: Odvajaj funkcijo $f(x) = \sin(x) + 3x$.

```
In[82]:= Clear[f]
In[83]:= f[x_] = Sin[x] + 3 x
Out[83]= 3 x + Sin[x]
In[84]:= f '[x]
Out[84]= 3 + Cos[x]
In[85]:= ∂x (Sin[x] + 3 x)
Out[85]= 3 + Cos[x]
```

Višje odvode računamo s podobnimi ukazi. Drugi odvod funkcije izračunamo z **f ''[x_]**, tretji odvod z **f '''[x_]** in tako dalje. *N*-ti odvod nam poda **D[f[x], {x, n}]** ali $\partial_{\{x,\,n\}}$ **f[x]**.

Zgled 3.29: Izračunaj tretji odvod funkcije $g(x) = \cos(x)$, šesti odvod od e^{4x} in tretji odvod od x^4.

```
In[86]:=  g[x_] = Cos[x]
Out[86]=  Cos[x]
In[87]:=  g'''[x_]
Out[87]=  Sin[x_]
In[88]:=  D[E^(4 x), {x, 6}]
Out[88]=  4096 e^(4 x)
In[89]:=  ∂{x,3} x^4
Out[89]=  24 x
```

Odvode funkcij več spremenljivk (parcialne odvode) po eni sami spremenljivki računamo z ukazi enakimi prejšnjim. Mešani odvod drugega reda dobimo z **D[f[x], x, y]** oziroma $\partial_{x,y}$ **f[x, y],** višje mešane odvode pa s **D[f[x, y], {x, n}, {y, m}]** ali $\partial_{\{x,n\},\{y,m\}}$ **f[x, y]**. Z *n* in *m* povemo število odvajanj po *x* oziroma po *y*.

Zgled 3.30: Odvajaj funkcijo

a) $\cos(x \cdot y)$ najprej po spremenljivki *x* in nato po *y*,
b) $x^3 \cdot y^5$ dvakrat po spremenljivki *x* in trikrat po *y*.

```
In[90]:=  D[Cos[x × y], x, y]
Out[90]=  -x y Cos[x y] - Sin[x y]
In[91]:=  ∂{x,2},{y,3} (x^3 × y^5)
Out[91]=  360 x y^2
```

Pri parcialnem odvajanju po več spremenljivkah zaporedoma je potrebno v argument zapisati zaporedje teh spremenljivk.

Zgled 3.31: Funkcijo $\sin(3\,x)\cdot\cos^2(y)$ odvajaj najprej dvakrat po spremenljivki *x*, nato pa še po *y*.

In[92]:= **D[Sin[3 x] × Cos[y]^2, x, x, y]**

Out[92]= 18 Cos[y] Sin[3 x] Sin[y]

3.7. INTEGRALNI RAČUN

Ukaz za integriranje po spremenljivki *x* je **Integrate[f[x_], x]**. S pomočjo palet lahko zapišemo tudi $\int$ ***f*** **[x_]** d***x***. Pri večkratnih integralih (dvojnih in trojnih) funkcije več spremenljivk moramo navesti vrstni red integriranja. To storimo z **Integrate[f[x_,y_, ...], x, y, ...]** ali $\int\int$**…f[x_, y_, ...]** d**x** d**y ...** .

Zgled 3.32: Izračunaj nedoločeni integral funkcije

a) $\frac{x+1}{x\,(x-1)^3}$ in

b) $-8\cos(3\,x)\sin(2\,y)\sin(z)$ po *z*, *y* ter *x* na oba načina.

In[93]:= **Integrate[(x + 1)/(x (x - 1)^3), x]**

Out[93]= $\frac{-2+x}{(-1+x)^2}+\text{Log}[-1+x]-\text{Log}[x]$

In[94]:= **Integrate[-8 Cos[3 x] Sin[2 y] Sin[z], z, y, x]**

Out[94]= $-\frac{4}{3}\,\text{Cos}[2\,y]\,\text{Cos}[z]\,\text{Sin}[3\,x]$

ali

In[95]:= $\int\int\int(-8\,\text{Cos}[3\,x]\,\text{Sin}[2\,y]\,\text{Sin}[z])\,dz\,dy\,dx$

Out[95]= $-\frac{4}{3}\,\text{Cos}[2\,y]\,\text{Cos}[z]\,\text{Sin}[3\,x]$

Integral funkcije med vrednostima a in b po spremenljivki x nam poda **Integrate[f[x_], {x, a, b}]** oziroma $\int_a^b f[x\_]\, dx$ v paletah. Določene integrale funkcije z več spremenljivkami izračunamo z **Integrate[f[x_,y_,...], {x, x_{min}, x_{max}}, {y, y_{min}, y_{max}}, ...]** ali $\int_{x_{min}}^{x_{max}} \int_{y_{min}}^{y_{max}} \ldots f[x\_,\ y\_,\ \ldots]\, dy\, dx$ ....

Zgled 3.33: Izračunaj

a) določeni integral funkcije $x^3 + \sin(x+1)$ na intervalu $x \in [-1,\ 4]$,

b) določeni integral funkcije $x^3 + 3$ na območju $x \in [0,\ 2]$, $y \in [x,\ 2x]$ in $z \in \left[0,\ x^2 + y^2 + 1\right]$,

c) integral $\int_0^{2\pi} \int_0^2 \int_0^{4-r^2} r\, z\, dz\, dr\, d\varphi$.

```
In[96]:= Integrate[x^3 + Sin[x + 1], {x, -1, 4}]
```

Out[96]= $\frac{259}{4} - \mathrm{Cos}[5]$

```
In[97]:= Integrate[x^3 + 3, {x, 0, 2},
           {y, x, 2 x}, {z, 0, x^2 + y^2 + 1}]
```

Out[97]= $\frac{11902}{105}$

In[98]:= $\int_0^{2\pi} \int_0^2 \int_0^{4-r^2} r\, z\, dz\, dr\, d\varphi$

Out[98]= $\frac{32\pi}{3}$

Vseh določemih integralov ni mogoče izračunati analitično. Tedaj uporabimo ukaz **NIntegrate**. Sintaksa ukazov je podobna prejšnjim.

3.8 RISANJE KRIVULJ V PARAMETRIČNI OBLIKI

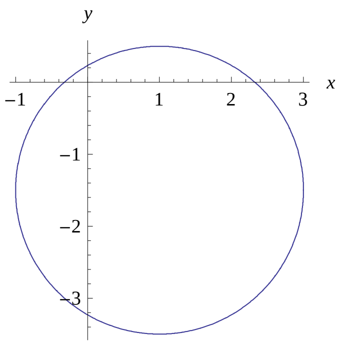
Krivulje v ravnini ali prostoru lahko podamo v parametrični obliki. To pomeni, da zapišemo koordinate vsake točke kot funkcijo nekega parametra. Ukaz za izris krivulje v tej obliki v ravnini je **ParametricPlot[{f_x, f_y}, {t, t_{min}, t_{max}}]**. Če želimo narisati več parametričnih krivulj v istem grafu, uporabimo **ParametricPlot[{{f_x, f_y}, {g_x, g_y},...}, {t, t_{min}, t_{max}}]**. Ukaz **AspectRatio→ Automatic** v spodnjem zgledu prilagodi velikost osi krožnici.

Zgled 3.34: Krožnico $(x-1)^2+\left(y+\frac{3}{2}\right)^2=4$, podano v kartezičnih koordinatah, zapiši v parametrični obliki in jo v tej obliki nariši.

Parametrična oblika krožnice: $(x,\ y)=\left(1+2\,\mathrm{Cos}\varphi,\ -\frac{3}{2}+2\,\mathrm{Sin}\varphi\right),\ \ \varphi\in[0,\,2\pi]$

In[99]:=
```
ParametricPlot[{1 + 2 Cos[φ], -3/2 + 2 Sin[φ]}, {φ, 0, 2 π},
  AxesLabel → {x, y}, AspectRatio → Automatic]
```

Out[99]=

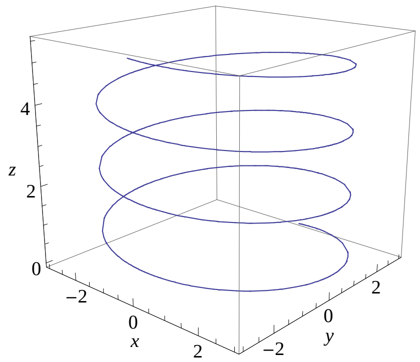


Krivuljo v prostoru narišemo podobno, in sicer z ukazom **ParametricPlot3D[{f_x, f_y}, {t, t_{min}, t_{max}}]**. Primer takšne krivulje je vijačnica.

Zgled 3.35: Nariši vijačnico$(x, y, z) = \left(3\,\mathrm{Cos}\varphi, 3\,\mathrm{Sin}\varphi, \frac{1}{4}t\right), \ t \in [0, 22]$.

```
In[100]:= ParametricPlot3D[{3 Sin[t], 3 Cos[t], 1/4 t},
   {t, 0, 22}, AxesLabel → {x, y, z}]
```

Out[100]=

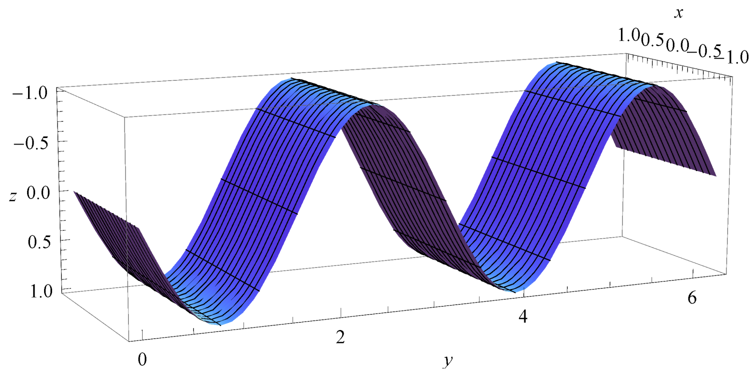


Poleg krivulj lahko v prostoru narišemo tudi ravnine. Ukaz je **ParametricPlot3D[{f_x, f_y, f_z}, {t, t_{min}, t_{max}},{u, u_{min}, u_{max}}].**

Zgled 3.36: Nariši ravnino $(z, t) = (z, t, \mathrm{Sin}(2\,t))$, $t \in [0, 2\pi]$ in $z \in [-1, 1]$.

```
In[101]:= ParametricPlot3D[{z, t, Sin[2 t]},
   {t, 0, 2 π}, {z, -1, 1}, AxesLabel → {x, y, z}]
```

Out[101]=

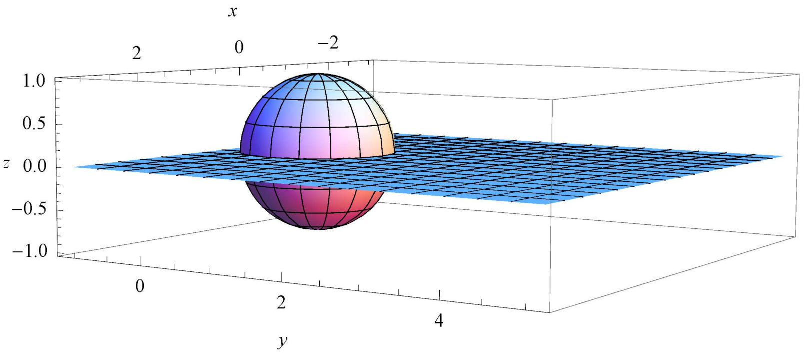


Več objektov v istem grafu narišemo s **ParametricPlot[{{f_x,f_y}, {g_x,g_y},...}, {t, t_{min}, t_{max}}]**.

Zgled 3.37: Nariši krožnico $(\theta, \varphi) = (\text{Sin}(\theta)\,\text{Cos}(\varphi),\ \text{Sin}(\theta)\,\text{Sin}[\varphi],\ \text{Cos}(\theta))$ in ravnino $(\theta, \varphi) = (\varphi - 3,\ 2 + \theta,\ 0)$ za $\theta \in [0, 2\pi]$ ter $\varphi \in [-\pi, \pi]$ v skupni koordinatni sistem.

In[102]:=
```
ParametricPlot3D[
 {{Sin[θ] Cos[φ], Sin[θ] Sin[φ], Cos[θ]},
  {φ - 3, 2 + θ, 0}}, {φ, 0, 2 π},
 {θ, -π, π}, AxesLabel → {x, y, z}]
```

Out[102]=



4 KOMBINATORIKA Z MATHEMATICO

V vsakdanjem življenju nas večkrat zanima, koliko elementov ima neka končna, neprazna množica in kateri so ti elementi. Prve probleme uvrščamo med probleme preštetja, probleme druge vrste pa k problemom naštetja. V primeru, da je množica manjša, lahko elemente z malo truda preštejemo ali jih zapišemo. V množici z veliko elementi, bi za enako nalogo potrebavali veliko več časa. Program *Mathematica* nam pri preštevanju in zapisu možnosti močno olajša delo.

Paket kombinatorike naložimo z ukazom **Needs["Combinatorica`"]**.

Zgled 4.1:

In[103]:= **Needs["Combinatorica`"]**

Pomembnejše funkcije, s katerimi se srečujemo pri reševanju kombinatoričnih nalog, so zapisane na spodnji sliki.

`n!`	factorial $n(n-1)(n-2)\times\ldots\times 1$
`n!!`	double factorial $n(n-2)(n-4)\times\ldots$
`Binomial[n,m]`	binomial coefficient $\binom{n}{m} = n!/[m!(n-m)!]$
`Multinomial[n1,n2,...]`	multinomial coefficient $(n_1+n_2+\ldots)!/(n_1!\,n_2!\ldots)$
`CatalanNumber[n]`	Catalan number c_n
`Hyperfactorial[n]`	hyperfactorial $1^1 2^2 \ldots n^n$
`BarnesG[n]`	Barnes G-function $1!\,2!\ldots(n-2)!$
`Subfactorial[n]`	number of derangements of n objects
`Fibonacci[n]`	Fibonacci number F_n
`Fibonacci[n,x]`	Fibonacci polynomial $F_n(x)$
`LucasL[n]`	Lucas number L_n
`LucasL[n,x]`	Lucas polynomial $L_n(x)$
`HarmonicNumber[n]`	harmonic number H_n
`HarmonicNumber[n,r]`	harmonic number H_n^r of order r
`BernoulliB[n]`	Bernoulli number B_n
`BernoulliB[n,x]`	Bernoulli polynomial $B_n(x)$
`NorlundB[n,a]`	Nörlund polynomial $B_n^{(a)}$
`NorlundB[n,a,x]`	generalized Bernoulli polynomial $B_n^{(a)}(x)$
`EulerE[n]`	Euler number E_n
`EulerE[n,x]`	Euler polynomial $E_n(x)$
`StirlingS1[n,m]`	Stirling number of the first kind $S_n^{(m)}$
`StirlingS2[n,m]`	Stirling number of the second kind $\mathcal{S}_n^{(m)}$
`BellB[n]`	Bell number B_n
`BellB[n,x]`	Bell polynomial $B_n(x)$
`PartitionsP[n]`	the number $p(n)$ of unrestricted partitions of the integer n
`IntegerPartitions[n]`	partitions of an integer
`PartitionsQ[n]`	the number $q(n)$ of partitions of n into distinct parts
`Signature[{i1,i2,...}]`	the signature of a permutation

Slika 7: Funkcije v kombinatoriki.

4.1 OSNOVE KOMBINATORIKE

Enostavne naloge lahko v kombinatoriki rešimo kar s pomočjo dveh preprostih pravil. To sta pravilo produkta oziroma osnovni izrek kombinatorike in pravilo vsote.

Pravilo produkta ali osnovni izrek kombinatorike

Pravilo produkta pravi, če lahko izberemo element $a \in A$ na n načinov in elemen $b \in B$ na m načinov, potem lahko izberemo urejeni par (a, b) na $n \cdot m$ načinov.

Posplošitev pravila na več elementov ($k \geqslant 2$) je očitna. Vlogo elementov imajo pogosto karte, žoge, števila, črke, knjige, ...

Zgled 4.2:

Za kosilo imamo na voljo 3 predjedi, 5 glavnih jedi in 2 sladici. Koliko različnih jedilnikov lahko sestavimo?

$$\mathbf{3 \times 5 \times 2}$$

$$30$$

Sestavimo lahko 30 različnih jedilnikov.

Pravilo vsote

Pravilo vsote pravi, če sta A in B končni disjunktivni množici, tedaj je $|A \cup B| = |A| + |B|$.

Pravilo lahko posplošimo za več množic, če so vse množice med seboj disjunktivne.

Zgled 4.3:

Saša ima v omari 4 športne čepice, 3 klobučke in 6 rutic. Na koliko načinov se lahko pokrije?

$$\mathbf{4 + 3 + 6}$$

$$13$$

Pokrije se lahko na 13 načinov.

Pravilo vsote in produkta pogosto nastopata skupaj.

Zgled 4.4:

Pohodnika želita prispeti iz kraja A v kraj E. Na zemljevidu vidita, da morata pri tem iti skozi mesta B, C ali D. Med krajema A in B so tri poti, med B in E pa dve. Od mesta A skozi mesto C ter dalje do E vodi ena sama pot. Med A in D imamo štiri poti, naprej do E pa vodijo samo tri poti. Med koliko različnimi potmi izbirata?

$$\mathbf{3 \times 2 + 1 \times 1 + 4 \times 3}$$

$$19$$

Izbirata med devetnajstimi različnimi potmi.

Urejene izbire in neurejene izbire

Problem si dostikrat ponazorimo kot izbiranje k predmetov iz množice z n predmeti. Izmed predmetov lahko izberemo nič, en, dva ali več predmetov. Z večanjem k-ja se število možnih izbir povečuje. Možnost izbir se poveča tudi, če predmete vračamo v začetno množico. Podrobneje poglejmo posamezne primere.

Urejene izbire najlažje opišemo s funkcijam.

IZREK 4.1: Naj bosta X in Y končni, neprazni množici in naj bo $F = \{f : X \rightarrow Y\}$. Tedaj je $|F| = |Y|^{|X|}$.

DOKAZ: Naj bo $|X| = \{x_1, x_2, ..., x_k\}$ in $|Y| = \{y_1, y_2, ..., y_n\}$. Vsakemu elementu iz množice X priredimo element iz množice Y. Pri tem lahko elementu x_1 priredimo kateregakoli izmed n elementov množice Y, elementu x_2 lahko zopet priredimo n elementov množice Y kakor tudi vsem nadaljnim elementom. Po pravilu produkta potem velja $|F| = n \cdot n \cdot ... \cdot n = n^k = |Y|^{|X|}$

Izrek 4.1 si lahko interpretiramo tudi drugače:

Število urejenih izbir s ponavljanjem k elementov iz množice z n elementi (n-množice) je enako n^k.

OPOMBA 4.2: Če sta X in Y končni množici in je $|X| = |Y|$, tedaj med njima obstaja bijekcija (in obratno).

IZREK 4.3: Naj bosta X, Y končni množici in $F = \{f : X \to Y;\ f \text{ je injektivna}\}$. Tedaj je $|F| = n \cdot (n-1) \cdot (n-2) \cdot ... \cdot (n-k+1)$, kjer je $|X| = k$ in $|Y| = n$.

DOKAZ: Naj bo $|X| = \{x_1, x_2, ..., x_k\}$ in $|Y| = \{y_1, y_2, ..., y_n\}$. Velja naj pogoj $n \geq k$, saj v obratnem primeru bijekcija ni možna. Vsakemu elementu iz množice X priredimo element iz množice Y. Pri tem lahko elementu x_1 priredimo kateregakoli izmed n elementov množice Y. Elementu x_2 lahko zaradi injektivnosti priredimo en element manj, in sicer $n-1$ elementov množice Y. Elementu x_3 lahko priredimo še en element manj, in sicer $n-2$ elementov, itd., zadnjemu elementu x_k pa lahko priredimo le $n-(k-1)$ elementov množice Y. Po pravilu produkta potem velja $|F| = n \cdot (n-1) \cdot (n-2) \cdot ... \cdot (n-k+1)$.

Izrek 4.3 si lahko zopet drugače interpretiramo:
Število urejenih izbir brez ponavljanja k elementov iz množice z n elementi (n-množice) je enako $(n)_k = n \cdot (n-1) \cdot (n-2) \cdot ... \cdot (n-k+1)$.

DEFINICIJA 4.4: Permutacija končne, neprazne množice X je bijekcija $X \to X$.

POSLEDICA 4.5: Število vseh permutacij v množici X z n elementi je enako
$n! = n \cdot (n\text{-}1) \cdot (n\text{-}2) \cdot ... \cdot 2 \cdot 1$.

DOKAZ: Vemo, da je permutacija bijektivna preslikava iz $X \to X$, zato je tudi injektivna. Po izreku 4.3 sledi $(n)_n = n \cdot (n-1) \cdot ... \cdot 2 \cdot 1$.

<u>Neurejene izbire</u> (s ponavljanjem in brez) k elementov izmed n elementov najlažje predstavimo z množicami.

Pravimo, naj bo X množica z n elementi. Zanimajo nas vse k-podmnožice množice X. Število teh nam poda binomsko število $\binom{n}{k}$.

DEFINICIJA 4.6: $\binom{n}{k} = |\{\, Y \subseteq X;\ |Y| = k,\ |X| = n\}|$.

To lahko še drugače zapišemo takole: Število neurejenih izbir brez ponavljanja dolžine k iz n-množice je $\binom{n}{k}$.

TRDITEV 4.7:

- $\binom{n}{0} = 1$, $\binom{n}{n} = 1$, $\binom{n}{1} = n$,
- $\binom{n}{k} = 0$, za $n \prec k$,

TRDITEV 4.8: Za $0 \leqslant k \leqslant n$ velja $\binom{n}{k} = \binom{n}{n-k}$.

DOKAZ: Naj bo X n-množica in Y njena k-podmnožica. Z Y^c označimo komplement množice Y glede na množico X. Zanima nas, koliko je množic Y in njenih komplementov. Vemo, da vsaki množici Y pripada natanko en komplement glede na množico Y. Ker je množici Y natanko $\binom{n}{k}$ in njenih komplementov natanko $\binom{n}{n-k}$, zaradi bijektivne preslikave med Y in Y^c velja $\binom{n}{k} = \binom{n}{n-k}$.

IZREK 4.9: Za $1 \leqslant k \leqslant n$ velja $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$.

DOKAZ:
Naj bo X množica z n elementi. Izberimo poljuben $x \in X$ in ga fiksirajmo. Tedaj preštejmo vse k-podmnožice v množici X. Ločimo dve možnosti:
i) k-podmnožica vsebuje element x,
ii) k-podnmožica ne vsebuje element x.
Število podmnožic je enako številu izbir elementov podmnožice. V prvem primeru izbiramo elemente podmnožice, pri čemer imamo element x že izbran. Preostale elemente v k-podmnožici določimo tako, da izmed preostalih $n-1$ elementov množice X izberemo $k-1$ elementov. Podmnožic je tedaj $\binom{n-1}{k-1}$. V drugem primeru moramo izbrati vse elemente k-podmnožice. Izbiramo lahko med $n-1$ elementi, saj elementa x ne smemo izbrati.Podmnožic je tedaj $\binom{n-1}{k}$. Po pravilu vsote je vseh podmnožic $\binom{n-1}{k-1}+\binom{n-1}{k}$.

IZREK 4.10: Za $0 \leqslant k \leqslant n$ velja $\binom{n}{k} = \frac{n!}{k! \cdot (n-k)!}$.

DOKAZ: Indukcija po n.

$n = 1$: $$\binom{1}{k} = \frac{1!}{k! \cdot (1-k)!}$$

$k = 0$: $$\binom{1}{0} = \frac{1!}{0! \cdot 1!}$$
$$1 = 1$$

$k = 1$: $$\binom{1}{1} = \frac{1!}{1! \cdot 0!}$$
$$1 = 1$$

$n \mapsto n+1$: $$\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k} =$$
$$= \frac{n!}{(k-1)! \cdot (n-(k-1))!} + \frac{n!}{k! \cdot (n-k)!} =$$
$$= \frac{n!}{(k-1)! \cdot (n-k+1)!} + \frac{n!}{k! \cdot (n-k)!} =$$
$$= \frac{n! \cdot k + n! \cdot (n-k+1)}{k! \cdot (n-k+1)!} = \frac{n! \cdot (k+n-k+1)}{k! \cdot (n-k+1)!} =$$
$$= \frac{(n+1)!}{k! \cdot ((n+1)-k)!}$$

IZREK 4.11: Število neurejenih izbir s ponavljanjem dolžine k iz n-množice je $\binom{n+k-1}{k}$.

DOKAZ: Naj bo X množica $\{x_1, x_2, ..., x_n\}$. Iz množice X izberemo k elementov, pri čemer dovolimo ponavljanje. Ker nas zanimajo neurejene izbire, si lahko vsak tak izbor predstavljamo kot:

$$x_1\, x_1 \ldots x_1 \quad x_2\, x_2 \ldots x_2 \quad \ldots \quad x_n\, x_n \ldots x_n$$

Pri tem x_i zapišemo k_i-krat ($0 \leq k_i \leq k$), če smo x_i izbrali k_i-krat. Temu zapisu priredimo binarno zaporedje ničel in enic. Z enicami nadomestimo x_i-je, z ničlami pa zapišemo mesta, kjer se stikata x_i in x_{i+1}. Prav tako zapišemo ničlo namesto nekega x_i, ki ga nismo izbrali (na primer, če v izboru nimamo x_4, zapišemo ničlo).

$$11...1011...10...011...1$$

Vseh neurejenih izbir s ponavljanjem bo toliko, kolikor je različnih zaporedij. Različnih zaporedij pa je toliko, kolikor je v zaporedju različnih načinov za razporeditev ničel oziroma enic. Če v zaporedju določimo mesta za ničle, so s tem mesta za enice že določena in obratno. V tem binarnem zaporedju imamo k enic in $n-1$ ničel. Skupaj imamo $n+k-1$ mest zaporedja. Ničle lahko poljubno razporedimo na katerokoli od $n+k-1$ mest zaporedja, kar zapišemo kot $\binom{n+k-1}{n-1}$. Z upoštevanjem trditve 4.7 velja še:

$$\binom{n+k-1}{n-1} = \binom{n+k-1}{(n+k-1)-(n-1)} = \binom{n+k-1}{k}.$$

Število izbir za posamezne dogodke nam podaja spodnja tabela.

$\ddots$	Urejene	Neurejene
S ponavljanjem	n^k	$\binom{n+k-1}{k}$
Brez ponavljanja	$(n)_k$	$\binom{n}{k}$

4.2. UKAZI V KOMBINATORIKI

4.2.1. FAKULTETA

Fakulteto števila n v *Mathematici* izračunamo s **Factorial[n]** ali **n!**. Možnost imamo izračunati tudi večkratne fakultete kot **Factorial2[n]** ali **n!!**, **n!!!**, itd. Dvojno fakulteto definiramo kot $n!! = n \cdot (n\text{-}2) \cdot (n\text{-}4) \cdot ... \cdot 5 \cdot 3 \cdot 1$. Pri tem moramo paziti, ker ukaza n!! ali n!!! nista enaka kot (n!)! ali ((n!)!)!.

<u>Zgled 4.5:</u> Izračunaj 20!, 5!! in (5!)!.

```
In[104]:= 20!
Out[104]= 2 432 902 008 176 640 000
In[105]:= 5!!
Out[105]= 15
In[106]:= (5!)! // Short
Out[106]//Short=
    66 895 029 134 491 270 575 881 180 540 903 725 867 527 463 331 ⋱
      380 298 102 956 713 523 <<68>>
     121 859 898 481 114 389 650 005 964 960 521 256 960 000 000 ⋱
      000 000 000 000 000 000 000
```

Če želimo izračunati fakultete za več števil, si pomagamo z ukazom **Table[n,{n,n$_{\mathbf{max}}$}]**. Pri tem *Mathematica* izračuna vrednosti za vsa naravna števila do $n_{\max}$.

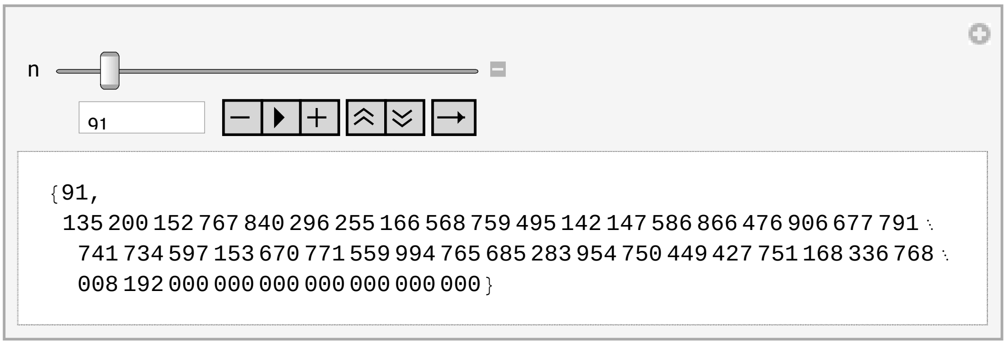
<u>Zgled 4.6:</u> Izračunaj fakultete števil {1, 2, ..., 7}.

```
In[107]:= Table[n!, {n, 7}]
Out[107]= {1, 2, 6, 24, 120, 720, 5040}
```

Fakulteto števila lahko zapišemo tudi z ukazom **Manipulate**, ki ima več oblik. Eno izmed njih smo uporabili v zgledu 4.7. V prvem zavitem oklepaju smo določili, da se naj vsakič izpiše število n in pripadajoča fakulteta. Z drugim zavitim oklepajem pa spreminjanje števila n. Tako s spreminjanjem drsnika izbiramo med števili od 1 do 1000, pri čemer veljajo samo vmesna cela števila (korak spreminjanja je 1).

Zgled 4.7:

In[108]:= **Manipulate[{n, n!}, {n, 1, 1000, 1}]**

Out[108]=


Izraze s fakultetami poenostavimo z **FullSimplify[izraz]**.

Zgled 4.8: Poenostavi izraz $\frac{(n+2)!}{n!!}$.

In[109]:= **FullSimplify[(n + 2)!! / n!!]**

Out[109]= 2 + n

4.2.2. BINOMSKO IN MULTINOMSKO ŠTEVILO

Binomsko število

V *Mathematici* nam binomsko število $\binom{n}{k}$ izračuna ukaz **Binomial [n,k]**.

Zgled 4.9: Izračunaj $\binom{10}{3}$.

In[110]:= **Binomial[10, 3]**

Out[110]= 120

ali

In[111]:= $\dfrac{\mathbf{10!}}{\mathbf{3!\,(10-3)!}}$

Out[111]= 120

V zgledu 4.10 vidimo, da nam tradicionalno obliko binomskega števila izpiše ukaz // **TraditionalForm**.

Zgled 4.10:

```
In[120]:= Binomial[n, k] // TraditionalForm
Out[120]//TraditionalForm=
```

$$\binom{n}{k}$$

Tedaj, ko želimo izračunati binomska števila z enakim *n*-jem in različnimi *k*-ji, zapišemo *k*-je v zaviti oklepaj. Podobno velja za več *n*-jev.

Zgled 4.11: Izračunaj $\binom{2}{3}, \binom{3}{3}, \binom{5}{3}, \binom{7}{3}$ in $\binom{11}{3}$.

```
In[121]:= Binomial[{2, 3, 5, 7, 11}, 3]
Out[121]= {0, 1, 10, 35, 165}
```

Ukaz **FullSimplify[izraz]** nam pri računanju s binomskimii koeficienti vrne najpreprostejšo obliko izraza.

Zgled 4.12: Poenostavi $\binom{n-1}{k-1} + \binom{n-1}{k}, \binom{n}{3}$ in $\frac{\binom{n+1}{k+1}}{\binom{n}{k}}$.

```
In[122]:= FullSimplify[Binomial[n - 1, k - 1] + Binomial[n - 1, k]]
Out[122]= Binomial[n, k]
In[123]:= Binomial[n, 3]
```

Out[123]= $\frac{1}{6}(-2+n)(-1+n)\,n$

```
In[124]:= FullSimplify[Binomial[n + 1, k + 1] / Binomial[n, k]]
```

Out[124]= $\frac{1+n}{1+k}$

Pascalov trikotnik podamo s pomočjo $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$ (za $1 \leqslant k \leqslant n$),

$\binom{n}{0} = 1$ in $\binom{n}{n} = 1$. V *Mathematici* pa ga skonstruiramo kot je prikazano v zgledu 4.13.

Zgled 4.13:

```
In[125]:= Column[
  Table[Binomial[n, k], {n, 0, 5}, {k, 0, n}], Center]

                {1}
              {1, 1}
            {1, 2, 1}
Out[125]= {1, 3, 3, 1}
          {1, 4, 6, 4, 1}
       {1, 5, 10, 10, 5, 1}
```

Multinomsko število

DEFINICIJA 4.12: Multimnožica je "množica", v kateri je dovoljeno ponavljanje.

Zgled 4.14: Razlika med multimnožico in množico.

{1, 1, 2, 2, 3} je multimnožica

{1, 2, 3} je množica

DEFINICIJA 4.13: Permutacija multimnožice moči *n* je urejena *n*-terica njenih elementov.

Zgled 4.15: Primera permutacije multimnožice {1, 2, 2, 3, 1}.

{1, 2, 2, 3, 1} in {1, 2, 3, 2, 1}

IZREK 4.14: Naj bo *M* multimnožica, ki vsebuje *k* različnih elementov, vsakega po n_i-krat, i = 1, 2, ..., k. (Seveda je $n_1 + n_2 + ... + n_k = n$.) Tedaj je število permutacij multimnožice *M* enako

$$\frac{n!}{n_1! \cdot n_2! \cdot ... \cdot n_k!}.$$

DOKAZ:

Za prvi element n_1 imamo na razpolago $\binom{n}{n_1}$ mest,

za drugi element n_2 imamo na razpolago $\binom{n-n_1}{n_2}$ mest, ...,

za k-ti element n_k imamo na razpolago $\binom{n-n_1-n_2-\ldots-n_{k-1}}{n_k}$ mest.

Po pravilu produkta je vseh načinov

$$\binom{n}{n_1}\cdot\binom{n-n_1}{n_2}\cdot\ldots\cdot\binom{n-n_1-n_2-\ldots-n_{k-1}}{n_k}=$$

$$=\frac{n!}{n_1!\cdot(n-n_1)!}\cdot\frac{(n-n_1)!}{n_2!\cdot(n-n_1-n_2)!}\cdot\frac{(n-n_1-n_2)!}{n_3!\cdot(n-n_1-n_2-n_3)!}\cdot\ldots\cdot\frac{(n-n_1-\ldots-n_{k-1})!}{n_k!\cdot(n-n_1-\ldots-n_k)!}=$$

$$=\frac{n!}{n_1!\cdot n_2!\cdot\ldots\cdot n_k!\cdot(n-(n_1+n_2!+\ldots+n_k))!}=\frac{n!}{n_1!\cdot n_2!\cdot\ldots\cdot n_k!}.$$

Števila iz izreka 4.13 označimo

$$\binom{n}{n_1,\ n_2,\ \ldots,\ n_k}=\frac{n!}{n_1!\cdot n_2!\cdot\ldots\cdot n_k!}$$

in jih imenujemo multinomska števila ali multinomski koeficienti. Multinomsko število v *Mathematici* izračuna ukaz **Multinomial[$\mathbf{n_1}$, $\mathbf{n_2}$, …, $\mathbf{n_k}$]**.

Zgled 4.16: Izračunaj $\binom{9}{2,\ 3,\ 4}$.

In[126]:= **Multinomial[2, 3, 4]**

Out[126]= 1260

V zgledu 4.17 nam tradicionalno obliko multinomskega števila izpiše:

Zgled 4.17:

In[127]:= **Multinomial[n, m, p] // TraditionalForm**

Out[127]//TraditionalForm=

$(m+n+p;\ m,\ n,\ p)$

IZREK 4.15 (Multinomski izrek): Naj bo $n \geq 1$ in $k \geq 1$. Tedaj velja

$$(a_1 + a_2 + ... + a_k)^n = \sum \binom{n}{n_1,\ n_2,\ ...,\ n_k} a_1^{n_1} a_2^{n_2} \cdot ... \cdot a_k^{n_k}.$$

Pri tem seštevamo po vseh urejenih k-tericah $(n_1,\ n_2,\ ...,\ n_k)$, za katere je $n_i \geq 0$ $(i = 1,\ 2,\ ...,\ k)$ in je $n_1 + n_2 + ... + n_k = n$.

DOKAZ:
$(a_1 + a_2 + ... + a_k)^n = (a_1 + a_2 + ... + a_k) \cdot ... \cdot (a_1 + a_2 + ... + a_k)$

Produkt je enak vsoti vseh možnih produktov, kjer iz vsakega izmed n oklepajev izberemo po en člen. Tipičen tak produkt je oblike $p = a_1^{n_1} a_2^{n_2} \cdot ... \cdot a_k^{n_k}$, kjer so $n_i \geq 0$ in velja $n_1 + n_2 + ... + n_k = n$. Tak produkt p dobimo tolikokrat, na kolikor načinov lahko izberemo:

- n_1 oklepajev za a_1: $\binom{n}{n_1}$,
- n_2 oklepajev za a_2: $\binom{n - n_1}{n_2}$, itd.,
- n_k oklepajev za a_k: $\binom{n - n_1 - ... - n_{k-1}}{n_k}$.

Po pravilu produkta je vseh načinov

$$\binom{n}{n_1} \cdot \binom{n - n_1}{n_2} \cdot ... \cdot \binom{n - n_1 - n_2 - ... - n_{k-1}}{n_k}.$$

Po kratkem izračunu zapisanem v dokazu izreka 4.14 dobimo $\frac{n!}{n_1! \cdot n_2! \cdot ... \cdot n_k!}$.

Zgled 4.18: Določi koeficient pred členom $x^2 \cdot y^2 \cdot z$ v izrazu $(x + y + z)^5$.

```
In[128]:= Multinomial[2, 2, 1]
Out[128]= 30
```

Razširimo $(x + y + z)^5$ in preverimo:

```
In[129]:= Expand[(x + y + z)^5]
```

Out[129]= $x^5 + 5 x^4 y + 10 x^3 y^2 + 10 x^2 y^3 + 5 x y^4 + y^5 + 5 x^4 z + 20 x^3 y z + 30 x^2 y^2 z + 20 x y^3 z + 5 y^4 z + 10 x^3 z^2 + 30 x^2 y z^2 + 30 x y^2 z^2 + 10 y^3 z^2 + 10 x^2 z^3 + 20 x y z^3 + 10 y^2 z^3 + 5 x z^4 + 5 y z^4 + z^5$

Vemo, da je število neurejenih izbir s ponavljanjem k elementov izmed n elementov enako $\binom{n+k-1}{k}$ (izrek 4.10). Po izreku 4.9 velja $\binom{n+k-1}{k} = \frac{(n+k-1)!}{(n-1)!\,k!}$. Zadnji zapis si lahko predstavljamo tudi kot multinomsko število.

<u>Zgled 4.19:</u> Izračunaj število neurejenih izbir s ponavljanjem štirih elementov izmed sedmih elementov na tri možne načine.

```
In[130]:= Binomial[7 + 4 - 1, 4]
Out[130]= 210
```

In[131]:= $\frac{(7+4-1)!}{(7-1)!\,4!}$

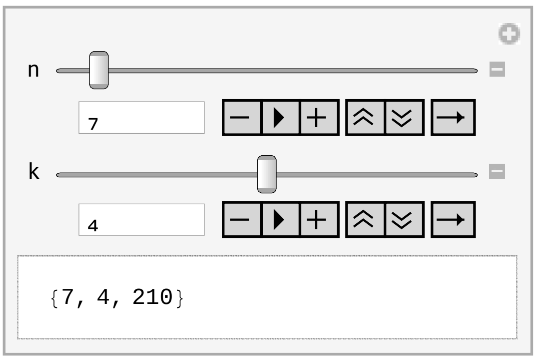
```
Out[131]= 210
In[132]:= Multinomial[7 - 1, 4]
Out[132]= 210
```

Z naslednjim ukazom prikažemo prikaz, ki nam izpiše število urejenih izbir s ponavljan jem za izbrani števili (zgled 4.18.). S prvim drsnikom izbiraš število vseh elementov n od 0 do 100. Z drugim drsnikom pa število k od 0 do n.

<u>Zgled 4.20:</u>

```
In[133]:= Table[Manipulate[{n, k, Multinomial[n - 1, k]},
             {n, 1, 100, 1}, {k, 1, n, 1}]]
```

Out[133]=



4.2.3. UREJENE VRSTE

Razvrstitve n elementov na k prostorov v vrsti s ponavljanjem podamo z ukazom **Tuples[{x, y,...}, k]**, kjer n elementov podamo v seznamu kot $\{x, y, \ldots\}$. Vse razvrstitve izračunamo z n^k.

<u>Zgled 4.21:</u> Zapiši vse razvrstitve elementov $\{j, k\}$ na 3 prostore v vrsti in jih preštej.

```
In[134]:= Tuples[{j, k}, 3]

Out[134]= {{j, j, j}, {j, j, k}, {j, k, j}, {j, k, k},
  {k, j, j}, {k, j, k}, {k, k, j}, {k, k, k}}

In[135]:= 2^3

Out[135]= 8
```

Če v ukazu Tuples podamo več urejenih vrst, *Mathematica* sestavi vse možne nove urejene vrste. Na prvem mestu v novi urejeni vrsti nastopa vedno element iz prve urejene vrste, na drugem mestu element iz druge urejene vrste in tako dalje. V primeru, da imamo v eni dani urejeni vrsti samo en element, bo ta nastopal v vseh novih urejenih vrstah.

<u>Zgled 4.22:</u> Po zgoraj opisanem načinu sestavi iz urejenih vrst $\{x, y, z\}$, $\{1, 2\}$ in $\{a\}$ novo urejeno vrsto.

```
In[136]:= Tuples[{{x, y, z}, {1, 2}, {a}}]

Out[136]= {{x, 1, a}, {x, 2, a}, {y, 1, a},
  {y, 2, a}, {z, 1, a}, {z, 2, a}}
```

4.2.4. PODMNOŽICE

DEFINICIJA 4.16: B je podmnožica množice A natanko tedaj, ko je vsak element množice B hkrati element množice A.

Množica z n elementi ima 2^n podmnožic vključno s samo sabo in prazno množico. Celotno število različnih k podmnožic množice z n elementi podamo z binomsko vsoto $\sum_{k=0}^{n}\binom{n}{k}=2^n$ oziroma

In[137]:= **Sum[Binomial[n, k], {k, 0, n}]**

Out[137]= 2^n

Vse podmnožice (subsets) dane množice se izpišejo z ukazom **Subset[seznam]**.

Zgled 4.23: Zapiši vse podmnožice množice {*a*, *b*, *c*}

In[138]:= **Subset[{a, b, c}]**

Out[138]= Subset[{a, b, c}]

Število podmnožic je v tem primeru enako:

In[139]:= $\mathbf{2^3}$

Out[139]= 8

K-podmnožica *B* je podmnožica množice *A* z *n* elementi ter vsebuje natanko *k* elementov. Število *k*-podmnožic z *n* elementi je zato podano z binomskim koeficientom $\binom{n}{k}$ oziroma **Binomial[n, k]**. *K*-podmnožice nekega seznama elementov podamo v *Mathematica* s **Subset[seznam, {k}].**

Zgled 4.24: Zapiši vse 2-podmnožice množice {*a*, *b*, *c*, *d*} in jih preštej.

In[140]:= **Subsets[{a, b, c, d}, {2}]**

Out[140]= {{a, b}, {a, c}, {a, d}, {b, c}, {b, d}, {c, d}}

In[141]:= **Binomial[4, 2]**

Out[141]= 6

4.2.5. PERMUTACIJE

DEFINICIJA 4.17: Permutacija je bijekcija iz $X \to X$, kjer je $X = \{1, 2, \ldots, n\}$ končna množica.

Permutacijo Π: $\{1, 2, \ldots, n\} \to \{1, 2, \ldots, n\}$ zapišemo v obliki

$$\Pi = \begin{pmatrix} 1 & 2 & 3 & \ldots & n-1 & n \\ \Pi(1) & \Pi(2) & \Pi(3) & \ldots & \Pi(n-1) & \Pi[n] \end{pmatrix}$$

Cikel imenujemo permutacijo naslednje oblike:

$$\Pi = \begin{pmatrix} k_1 & k_2 & k_3 & \ldots & k_{m-1} & k_m & l_1 & l_2 & \ldots & l_{n-m} \\ k_2 & k_3 & k_4 & \ldots & k_m & k_1 & l_1 & l_2 & \ldots & l_{n-m} \end{pmatrix} = (k_1 k_2 k_3 \ldots k_m).$$

Cikel $(k_1 k_2 k_3 \ldots k_m)$ je cikel dolžine m.

Množico vseh permutacij množice $\{1, 2, \ldots, n\}$ označimo s S_n.

Število vseh permutacij množice $\{1, 2, \ldots, n\}$ je enako $n!$.

Seznam vseh možnih permutacij množice $\{x, y, \ldots\}$ dobimo z ukazom **Permutations[{x, y, ...}]**. Če želimo permutacije prvih n naravnih števil, zadošča ukaz **Permutations[n]** ali **Permutations[Range[n]]**.

<u>Zgled 4.25:</u> Zapiši seznam vseh možnih permutacij elementov {1, 2, 3} na vse tri načine.

```
In[142]:= Permutations[{1, 2, 3}]
Out[142]= {{1, 2, 3}, {1, 3, 2}, {2, 1, 3},
           {2, 3, 1}, {3, 1, 2}, {3, 2, 1}}
In[143]:= Permutations[3]
Out[143]= {{1, 2, 3}, {1, 3, 2}, {2, 1, 3},
           {2, 3, 1}, {3, 1, 2}, {3, 2, 1}}
In[144]:= Permutations[Range[3]]
Out[144]= {{1, 2, 3}, {1, 3, 2}, {2, 1, 3},
           {2, 3, 1}, {3, 1, 2}, {3, 2, 1}}
```

Število permutacij dobimo tako, da prejšnjim ukazom dodamo ukaz **Length** ali izračunamo **n!**.

<u>Zgled 4.26:</u> Izračunaj število permutacij treh elementov na oba načina.

```
In[145]:= 3!
Out[145]= 6
In[146]:= Length[Permutations[3]]
Out[146]= 6
```

Število urejenih izbir brez ponavljanja k elementov izmed n elementov množice $\{x, y, \ldots\}$ izpiše ukaz **Permutations[{x,y,...},{k}]**, medtem ko jih prešteje $\mathbf{\frac{n!}{(n-k)!}}$ ali **Length[Permutations[{x,y,...}, {k}]]**.

Zgled 4.27: Zapiši vse urejene izbire brez ponavljanja dveh elementov iz elementov {*a*, *b*, *c*, *d*} in jih preštej.

```
In[147]:= Permutations[{a, b, c, d}, {2}]
Out[147]= {{a, b}, {a, c}, {a, d}, {b, a}, {b, c}, {b, d},
           {c, a}, {c, b}, {c, d}, {d, a}, {d, b}, {d, c}}
In[148]:= Length[Permutations[{a, b, c, d}, {2}]]
Out[148]= 12
In[149]:= 4! / (4 - 2)!
Out[149]= 12
```

Permutacije vseh možnih ciklov danih elementov množice {*x*, *y*, ...} izpiše **Permutations[{x,y,...}, All]**. Pri tem najprej izpiše najkrajše cikle. Ukaz **Permutations[{x,y,...}, n]** pa poda vse cikle danih elementov množice {*x*, *y*, ...} dolžine *n* ali manj.

Zgled 4.28: Zapiši vse cikle elementov {1, 2, 3} ter vse cikle elementov {*p*, *q*, *r*, *s*} dolžine 2 ali manj.

```
In[150]:= Permutations[{1, 2, 3}, All]
Out[150]= {{}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 1},
           {2, 3}, {3, 1}, {3, 2}, {1, 2, 3}, {1, 3, 2},
           {2, 1, 3}, {2, 3, 1}, {3, 1, 2}, {3, 2, 1}}
In[151]:= Permutations[{p, q, r, s}, 2]
Out[151]= {{}, {p}, {q}, {r}, {s}, {p, q}, {p, r},
           {p, s}, {q, p}, {q, r}, {q, s}, {r, p},
           {r, q}, {r, s}, {s, p}, {s, q}, {s, r}}
```

Ukaz **Permutations[{x,y,...}, {k_{min}, k_{max}, -1}]**, zapiše vse permutacije elementov množice {*x*, *y*, ...}, kjer je dolžina cikla med k_{max} in k_{min}. Ukaz lahko posplošimo na **Permutations[{x,y,...}, {k_{min}, k_{max}, -korak}]**. S parametrom imenovanim *korak* povemo, za koliko se manjša dolžina cikla od dolžine k_{max} proti k_{min}.

Zgled 4.29: Zapiši vse cikle

a) elementov {u, v, w} dolžine med nič in tri,

b) prvih treh naravnih števil dolžine med nič in tri, kjer je korak enak -2.

```
In[152]:= Permutations[{u, v, w}, {2, 0, -1}]

Out[152]= {{u, v}, {u, w}, {v, u}, {v, w},
           {w, u}, {w, v}, {u}, {v}, {w}, {}}

In[153]:= Permutations[{1, 2, 3}, {3, 0, -2}]

Out[153]= {{1, 2, 3}, {1, 3, 2}, {2, 1, 3},
           {2, 3, 1}, {3, 1, 2}, {3, 2, 1}, {1}, {2}, {3}}
```

Če želimo v zgornjih treh zgledih permutacije prvih n naravnih števil, zadošča namesto elementov množice **{*x*, *y*, ...}** pisati **Range[n]**.

V primeru, da se elementi v množici ponavljajo, dobimo t.i. multimnožico. Število permutacij multimnožice M dobimo z multinomskim simbolom. Vse možne permutacije multimnožice pa prikaže ukaz **Permutations** ali **DistinctPermutations**, ki ne loči med enakimi elementi.

Zgled 4.30: Zapiši in preštej vse možne razporeditve elementov {a, a, b, c}.

```
In[154]:= Multinomial[2, 1, 1]

Out[154]= 12

In[155]:= Permutations[{a, a, b, c}]

Out[155]= {{a, a, b, c}, {a, a, c, b}, {a, b, a, c}, {a, b, c, a},
           {a, c, a, b}, {a, c, b, a}, {b, a, a, c}, {b, a, c, a},
           {b, c, a, a}, {c, a, a, b}, {c, a, b, a}, {c, b, a, a}}
```

ali

```
In[156]:= DistinctPermutations[{a, a, b, c}]

Out[156]= {{a, a, b, c}, {a, a, c, b}, {a, b, a, c}, {a, b, c, a},
           {a, c, a, b}, {a, c, b, a}, {b, a, a, c}, {b, a, c, a},
           {b, c, a, a}, {c, a, a, b}, {c, a, b, a}, {c, b, a, a}}
```

4.2.6. FIBONACCIJEVA IN STIRLINGOVA ŠTEVILA

Fibonaccijeva števila

Leonardo Fibonacci ali Leonardo Pisano (1170-1250) je bil italijanski matematik. Po njem se imenuje Fibonaccijevo zaporedje. Sestavljajo ga t.i. Fibonaccijeva števila. To so števila določena z rekurzivno zvezo $F_n=F_{n-1}+F_{n-2}$, pri začetnem pogoju $F_0=F_1=1$.

Zgled 4.31: Izračunaj Fibonacijeva števila do člena $n = 15$.

In[157]:= **Table[Fibonacci[n], {n, 15}]**

Out[157]= {1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610}

V zgledu 4.32 zapišemo tradicionalno obliko Fibonaccijevih števil v *Mathematici*.

Zgled 4.32:

In[158]:= **Fibonacci[n] // TraditionalForm**

Out[158]//TraditionalForm=

F_n

DEFINICIJA 4.18: Fibonaccijev polinom definiramo z rekurzivno zvezo $F_{n+1}(x) = x\,F_n(x) + F_{n-1}(x)$, pri čemer je $F_1(x) = 1$ in $F_2(x) = x$.

Zgled 4.33: Prvih nekaj Fibonaccijevih polinomov.

$F_1(x) = 1$
$F_2(x) = x$
$F_3(x) = x^2 + 1$
$F_4(x) = x^3 + 2\,x$
$F_5(x) = x^4 + 3\,x^2 + 1$

Fibonaccijev polinom $F_n(x)$ izpiše ukaz **Fibonacci[n, x]**.

Zgled 4.34: Zapiši Fibonaccijev polinom $F_n(x)$ za $n = 15$.

In[159]:= **Fibonacci[15, x]**

Out[159]= $1 + 28\,x^2 + 126\,x^4 + 210\,x^6 + 165\,x^8 + 66\,x^{10} + 13\,x^{12} + x^{14}$

Stirlingova števila

Stirlingova števila se imenujejo po škotskem matematiku J.Stirlingu (1692-1770). Poznamo dve vrsti Stirlingovih števil, in sicer Stirlingova števila prve vrste in Stirlingova števila druge vrste.

DEFINICIJA 4.19: <u>Stirlingovo število prve vrste</u> $\left[{n \atop k}\right]$ je enako številu permutacij iz S_n, ki se zapišejo kot produkt k disjunktnih ciklov. Pri tem velja $1 \leq k \leq n$. Če je $n < k$, potem je $\left[{n \atop k}\right]=0$.

Rečemo lahko tudi, da nam Stirlingovo število prve vrste $\left[{n \atop k}\right]$ pove število načinov razporeditev n elementov v k ciklov.

TRDITEV 4.20: a) $\left[{n \atop n}\right]=1$

b) $\left[{n \atop 1}\right]=(n-1)!$

c) $\left[{n \atop n-1}\right] = \binom{n}{2}$

IZREK 4.21: $\left[{n \atop k}\right] = \left[{n-1 \atop k-1}\right] + (n-1)\cdot\left[{n-1 \atop k}\right]$, za $1 \leq k \leqslant n$.

DOKAZ: Naj bo Π poljubna permutacija iz S_n s k cikli. Permutacijo Π dobimo iz permutacije $\Pi^j \in S_{n-1}$ na enega od dveh načinov.

i) V Π^j dodamo n-ti element v svoj cikel.

Potem preostale elemente, ki jih je $n-1$, razvrstimo v preostalih $k-1$ ciklov. Permutacijo Π tedaj dobimo na $\left[{n-1 \atop k-1}\right]$ načinov.

ii) Π^j ima k ciklov, element n vrinemo v enega izmed njih.

Potem preostalih $n-1$ elementov razvrstimo v k ciklov. Element n pa lahko v k ciklov vrinemo na $n-1$ načinov.

Po pravilu vsote dobimo permutacijo Π na $\left[{n-1 \atop k-1} \right] + (n-1) \cdot \left[{n-1 \atop k} \right]$ načinov.

V Mathematici računamo Stirlingova števila prve vrste z ukazom **StirlingFirst[n, k]** ali **StirlingS1[n, k]** (od različice 6.0 dalje).

Zgled 4.35: Izračunaj $\left[{5 \atop 3} \right]$.

```
In[160]:= StirlingS1[5, 3]
Out[160]= 35
```

Isti primer bi "peš" reševali takole :

$$\left[{5 \atop 3} \right] = \left[{4 \atop 2} \right] + 4 \cdot \left[{4 \atop 3} \right] = \left(\left[{3 \atop 1} \right] + 3 \cdot \left[{3 \atop 2} \right] \right) + 4 \cdot \binom{4}{2} =$$

$$= 2! + 3 \cdot \binom{3}{2} + 4 \cdot \frac{4 \cdot 3}{2 \cdot 1} = 6 + 3 \cdot \frac{3 \cdot 2}{2 \cdot 1} + 24 = 35$$

IZREK 4.22 : $\sum_{k=1}^{n} \left[{n \atop k} \right] x^k = x \cdot (x+1) \cdot (x+2) \cdot \ldots \cdot (x+n-1),$

kjer nam $\left[{n \atop k} \right]$ poda koeficient pred x^k.

DOKAZ: Indukcija po n.

$n = 1$: $\left[{1 \atop 1} \right] x^1 = x$

$$x = x$$

$n \mapsto n+1$: Pričnimo na desni.

$x \cdot (x+1) \cdot (x+2) \cdot \ldots \cdot (x+n-2) \cdot (x+n-1) =$

$= x \cdot (x+1) \cdot (x+2) \cdot \ldots \cdot (x+(n-1)-1) \cdot (x+n-1) =$

$$= \left(\sum_{k=1}^{n-1} \left[{n-1 \atop k} \right] x^k\right) (x+(n-1)) =$$

$$= \sum_{k=1}^{n-1} \left[{n-1 \atop k} \right] x^{k+1} + \sum_{k=1}^{n-1} \left[{n-1 \atop k} \right] x^k (n-1) =$$

vpeljemo novo spremenljivko: $k' = k+1$

$k = 1 \to k' = 2$ in $k = n-1 \to k' = n$

$$= \sum_{k'=2}^{n} \left[{n-1 \atop k'-1} \right] x^{k'} + (n-1) \sum_{k=1}^{n-1} \left[{n-1 \atop k} \right] x^k =$$

pišemo $k = k'$

$$= \sum_{k=2}^{n} \left[{n-1 \atop k-1} \right] x^k + (n-1) \sum_{k=1}^{n-1} \left[{n-1 \atop k} \right] x^k =$$

$$= \sum_{k=1}^{n} \left[{n-1 \atop k-1} \right] x^k - \left[{n-1 \atop 0} \right] x + \sum_{k=1}^{n-1} (n-1) \left[{n-1 \atop k} \right] x^k =$$

$$= \sum_{k=1}^{n} x^k \left(\left[{n-1 \atop k-1} \right] + (n-1) \left[{n-1 \atop k} \right] \right) - (n-1) \left[{n-1 \atop n} \right] x^n =$$

$$= \sum_{k=1}^{n} x^k \left[{n \atop k} \right]$$

Zgled 4.36: Izračunaj koeficient pred x^4 v $q(x) = x(1+x)(2+x) \cdot \ldots \cdot (6+x)$.

```
In[161]:= StirlingFirst[7, 4]
Out[161]= 735
```

Pomnožimo $x \cdot (1+x) \cdot (2+x) \cdot \ldots \cdot (6+x)$ in preverimo:

```
In[162]:= Expand[x (1 + x) (2 + x) (3 + x) (4 + x) (5 + x) (6 + x)]
Out[162]= 720 x + 1764 x^2 + 1624 x^3 + 735 x^4 + 175 x^5 + 21 x^6 + x^7
```

Z upoštevanjem izreka 4.21 in robnih pogojev $\left[{n \atop n}\right]$ =1 ter $\left[{n \atop 1}\right] = (n-1)!$ zapišemo trikotnik Stirlingovih števil prve vrste (zgled 4.37).

<u>Zgled 4.37:</u>

In[163]:=
```
Column[Table[StirlingFirst[n, k],
  {n, 0, 5}, {k, 0, n}], Center]
```

Out[163]=
```
        {1}
       {0, 1}
      {0, 1, 1}
     {0, 2, 3, 1}
   {0, 6, 11, 6, 1}
{0, 24, 50, 35, 10, 1}
```

Stirlingovo število prve vrste je med drugim enako številu vseh permutacij n elementov razporejenih v m ciklov.

<u>Zgled 4.38:</u> Izračunaj na koliko načinov lahko 6 gostov sede za 3 mize.

In[164]:=
```
NumberOfPermutationsByCycles[6, 3]
```

Out[164]= 225

ali

In[165]:=
```
StirlingFirst[6, 3]
```

Out[165]= 225

Sedejo lahko na 225 načinov.

DEFINICIJA 4.23: <u>Stirlingovo število druge vrste</u> $\left\{{n \atop k}\right\}$ je enako številu razbitij množice z n elementi na k nepraznih rezredov, če velja $1 \le k \le n$. Če je $n < k$, potem je $\left\{{n \atop k}\right\}$ =0

TRDITEV 4.24: a) $\left\{ {n \atop n} \right\} = 1$

b) $\left\{ {n \atop 1} \right\} = 1$

c) $\left\{ {n \atop n-1} \right\} = \binom{n}{2}$

IZREK 4.25: $\left\{ {n-1 \atop k-1} \right\} + k \cdot \left\{ {n-1 \atop k} \right\} = \left\{ {n \atop k} \right\}$

DOKAZ: Razbitje množice $\{1, 2, ..., n\}$ na k razredov lahko dobimo iz razbitja množice $\{1, 2, ..., n-1\}$ na enega izmed dveh načinov.

i) *N*-ti element damo v svoj razred.

Potem množico z $n-1$ elementi razbijemo v preostalih $k-1$ razredov. To storimo na $\left\{ {n-1 \atop k-1} \right\}$ načinov.

ii) *N*-ti element dodamo v enega izmed obstječih k razredov.

Potem množico z $n-1$ elementi razbijemo v k razredov. To storimo na $\left\{ {n-1 \atop k-1} \right\}$ načinov. Ker n dodamo v enega izmed obstječih k razredov, imamo zanj k možnosti. Število razbitij je tedaj enako $k \cdot \left\{ {n-1 \atop k-1} \right\}$.

Po pravilu vsote imamo skupaj $\left\{ {n-1 \atop k-1} \right\} + k \cdot \left\{ {n-1 \atop k} \right\}$ načinov razbitij n-množice na k razredov.

Ukaz **StirlingS2[n, k]** oziroma **StirlingSecond[n, k],** nam izračuna Stirlingova števila druge vrste.

Zgled 4.39: Izračunaj $\left\{ {6 \atop 3} \right\}$.

In[166]:= **StirlingS2[6, 3]**

Out[166]= 90

Glede na trditev 4.24 velja $\left\{ {n \atop n-1} \right\} = \binom{n}{2}$, kjer je desna stran binomsko število. V zgledu 4.40 preveri ali velja $\left\{ {5 \atop 4} \right\} = \binom{5}{2}$.

<u>Zgled 4.40:</u>

```
In[167]:= StirlingS2[5, 4]
Out[167]= 10
In[168]:= Binomial[5, 2]
Out[168]= 10
```

Vsa razbitija množice z n elementi na k nepraznih rezredov nam zapiše ukaz **KSetPartitions[n,k]**.

<u>Zgled 4.41:</u> Zapiši vsa razbitja množice s petimi elementi na štiri neprazne razrede.

```
In[169]:= KSetPartitions[5, 4]
Out[169]= {{{1}, {2}, {3}, {4, 5}}, {{1}, {2}, {3, 4}, {5}},
  {{1}, {2}, {3, 5}, {4}}, {{1}, {2, 3}, {4}, {5}},
  {{1}, {2, 4}, {3}, {5}}, {{1}, {2, 5}, {3}, {4}},
  {{1, 2}, {3}, {4}, {5}}, {{1, 3}, {2}, {4}, {5}},
  {{1, 4}, {2}, {3}, {5}}, {{1, 5}, {2}, {3}, {4}}}
```

Z izrekom 4.25 in robnima pogojema $\left\{ {n \atop n} \right\} = 1$ ter $\left\{ {n \atop 1} \right\} = 1$ tvorimo trikotnik Stirlingovih števil druge vrste (zgled 4.42).

<u>Zgled 4.42:</u>

```
In[170]:= Column[Table[StirlingSecond[n, k],
   {n, 0, 5}, {k, 0, n}], Center]
                  {1}
                {0, 1}
              {0, 1, 1}
Out[170]=   {0, 1, 3, 1}
          {0, 1, 7, 6, 1}
      {0, 1, 15, 25, 10, 1}
```

4.2.7. PARTICIJE

DEFINICIJA 4.26: Particija pozitivnih naravnih števil je neurejena množica pozitivnih naravnih števil, katerih vsota je enaka številu n. Ta naravna števila se imenujejo sumandi ali deli od particije. Število particij števila n označimo kot $p(n)$.

<u>Zgled 4.43:</u> Število šest lahko zapišemo kot vsoto treh sumandov.
$6 = 4 + 1 + 1$

Z **Partitions[n]** zapišemo vse particije naravnega števila n, pri čemer je v prvi particiji največji sumand (t.i. leksikalni vrstni red). Isti odgovor nam vrne **IntegerPartitions[n]**. Število particij izvemo z **NumberOfPartitions[n]**, **PartitionsP[n]** ali **Length[IntegerPartitions[n]]**.

<u>Zgled 4.44:</u> Zapiši vse particije števila 6 in preštej število vseh particij števila 6.

```
In[171]:= IntegerPartitions[6]
Out[171]= {{6}, {5, 1}, {4, 2}, {4, 1, 1},
  {3, 3}, {3, 2, 1}, {3, 1, 1, 1}, {2, 2, 2},
  {2, 2, 1, 1}, {2, 1, 1, 1, 1}, {1, 1, 1, 1, 1, 1}}
In[172]:= NumberOfPartitions[6]
Out[172]= 11
```

Particije števila n, ki vsebujejo same različne sumande preštejemo z **PartitionsQ[n]**.

<u>Zgled 4.45:</u> Preštej število particij števila 6 s samimi različnimi sumandi.

```
In[173]:= PartitionsQ[6]
Out[173]= 4
```

Ukaz **Partitions[n,k]** zapiše vse particije naravnega števila n, kjer je največji neničelni sumand število k. Število takih particij dobimo z **Length[Partitions[n, k]].**

Zgled 4.46: Zapiši vse particije števila *6* z največjim neničelnim sumandom 3 in preštej število vseh takih particij.

```
In[174]:= Partitions[6, 3]
Out[174]= {{3, 3}, {3, 2, 1}, {3, 1, 1, 1}, {2, 2, 2},
  {2, 2, 1, 1}, {2, 1, 1, 1, 1}, {1, 1, 1, 1, 1, 1}}
In[175]:= Length[Partitions[6, 3]]
Out[175]= 7
```

Naj bo $P(n, k)$ število particij števila n z največ k deli. Tedaj število zapisov števila n z največ k deli poda **Length[IntegerPartitions[n,k]]**. Z ukazom **IntegerPartitions[n,k]** pa predstavimo vse take zapise.

Zgled 4.47: Zapiši vse particije števila 5 z največ 3 deli in preštej število vseh takih zapisov.

```
In[176]:= IntegerPartitions[5, 3]
Out[176]= {{5}, {4, 1}, {3, 2}, {3, 1, 1}, {2, 2, 1}}
In[177]:= Length[IntegerPartitions[5, 3]]
Out[177]= 5
```

Število zapisov števila n kot vsoto natanko k sumandov označujemo z $p_k(n)$. V *Mathematici* ga podamo z ukazom **Length[IntegerPartitions[n,{k}]]**, kjer nam **IntegerPartitions[n,{k}]** izpiše vse take zapise. Če želimo zapise števila n med k_{min} in k_{max} sumandi, potem to storimo z **IntegerPartitions[n, {k_{min}, k_{max}}]**.

Zgled 4.48: Zapiši in preštej vse zapise

a) števila 6 kot vsoto natanko 3 sumandov,

b) števila 7 kot vsoto dveh in treh sumandov.

```
In[178]:= IntegerPartitions[6, {3}]
Out[178]= {{4, 1, 1}, {3, 2, 1}, {2, 2, 2}}
In[179]:= Length[IntegerPartitions[6, {3}]]
Out[179]= 3
```

In[180]:= **IntegerPartitions[7, {2, 3}]**

Out[180]= {{6, 1}, {5, 2}, {5, 1, 1},
{4, 3}, {4, 2, 1}, {3, 3, 1}, {3, 2, 2}}

In[181]:= **Length[IntegerPartitions[7, {2, 3}]]**

Out[181]= 7

Z naslednjim ukazom pa zapišemo vse particije števila s točno določenimi sumandi (na primer s sumandi 1, 2 in 3).

Zgled 4.49: Zapiši vse particije števila 7 s sumandi 1, 2 in 3.

In[182]:= **IntegerPartitions[7, All, {1, 2, 3}]**

Out[182]= {{3, 3, 1}, {3, 2, 2}, {3, 2, 1, 1},
{3, 1, 1, 1, 1}, {2, 2, 2, 1}, {2, 2, 1, 1, 1},
{2, 1, 1, 1, 1, 1}, {1, 1, 1, 1, 1, 1, 1}}

Z ukazom **NextPartition[p]** podamo particijo, ki sledi particiji *p* v obratnem leksikalnem vrstnem redu. **RandomPartition[n]** pa nam določi naključno particijo števila *n*.

Zgled 4.50: Zapiši naslednjo particijo particije {3, 3} in poljubno particijo števila 6.

In[183]:= **NextPartition[{3, 3}]**

Out[183]= {3, 2, 1}

In[184]:= **RandomPartition[6]**

Out[184]= {2, 2, 2}

Ferrers-ov diagram je način opisa particije. Spodnji ukaz (zgled 4.51) nariše Ferrers-ov diagram za particijo {10, 8, 5, 5, 4, 1}, ki jo v naslednjem delu zgleda konjugira (zamenja stolpce in vrstice).

Zgled 4.51:

In[185]:= **FerrersDiagram[{10, 8, 5, 5, 4}]**

```
● ● ● ● ● ● ● ● ● ●

● ● ● ● ● ● ● ●

● ● ● ● ●

● ● ● ● ●

● ● ● ●
```

Out[185]=

In[186]:= **FerrersDiagram[TransposePartition[{10, 8, 5, 5, 4}]]**

```
●   ●   ●   ●   ●
●   ●   ●   ●   ●
●   ●   ●   ●   ●
●   ●   ●   ●   ●
●   ●   ●   ●
●   ●
●   ●
●   ●
●
●
```

Out[186]=

Naslednji ukaz **DurfeeSquare[p]**, kjer je *p* neka particija, poda stranico največjega kvadrata vsebovanega v Ferrers-ovem diagramu.

Zgled 4.52: Zapiši dolžino stranice največjega kvadrata vsebovanega v Ferrers-ovem diagramu particije {10, 8, 5, 5, 4}.

In[187]:= **DurfeeSquare[{10, 8, 5, 5, 4}]**

Out[187]= 4

5. TEORIJA GRAFOV Z MATHEMATICO

Paket kombinatorike naložimo z ukazom **Needs["Combinatorica`"]**.

In[188]:= **Needs["Combinatorica`"]**

5.1. OSNOVNI POJMI

Graf $G = (V\ (G), E(G))$ sestavljata neprazna množica elementov, ki jih imenujemo vozlišča grafa, in seznam (neurejenih) parov teh elementov, ki jih imenujemo povezave grafa. Množico vozlišč grafa označimo z $V(G)$, seznam povezav pa z $E(G)$. Če sta u in v vozlišči grafa G, potem za povezavi uv ali vu rečemo, da povezujeta vozlišči u in v.

Dve ali več povezav, ki povezujejo isti par točk, poimenujemo vzporedne povezave. Povezava, ki povezuje neko točko s seboj, je povratna povezava ali zanka. Graf brez zank in večkratnih povezav poimenujemo enostavni graf.

Naj bo G graf brez zank, v pa vozlišče grafa G. Stopnja vozlišča v je število povezav, ki vsebujejo v. Stopnjo vozlišča označimo z deg v. Pravimo, da je graf regularen, če imajo vsa vozlišča grafa enako stopnjo. Če imajo vsa vozlišča grafa stopnjo r, je graf r-regularen.

Naj bosta u in v vozlišči grafa. Če sta u in v povezani s povezavo, potem rečemo, da sta sosednji in pišemo $e = uv$. Povezavi s skupnim krajiščem sta incidentni povezavi.

Sprehod dolžine k v grafu G je zaporedje k povezav grafa G oblike uv, vw, wx, …, yz. Tak sprehod označimo z $uvwx…yz$ in ga imenujemo sprehod med vozliščema u in z. Če so vse povezave sprehoda različne, potem sprehod imenujemo enostavni sprehod ali sled. Če so v enostavnem sprehodu vsa vozlišča različna, potem sprehod imenujem pot.

Sklenjen sprehod ali obhod v grafu G je zaporedje povezav grafa G oblike uv, vw, wx, …, yz, zu. Če so vse povezave obhoda različne, potem ga imenujemo enostavni obhod ali sklenjena sled. Če so v obhodu vse povezave in vsa vozlišča različna, potem ga poimenujemo cikel.

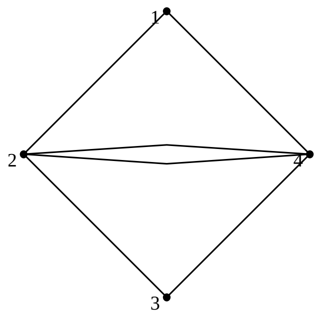
Graf G je povezan, če obstaja pot med poljubnim parom vozlišč, sicer je nepovezan.

Naj bo G graf brez zank z n vozlišči, označenimi z 1, 2, 3, ..., n. Matrika sosednosti M(G) je matrika razsežnosti $n \times n$, v katerem element v j-tem stolpcu i-te vrstice pove število povezav, ki povezujejo vozlišči i in j.

<u>Zgled 5.1:</u> Zapiši matriko sosednosti za spodnji graf.

In[189]:=
```
ShowGraph[AddEdges[Cycle[4], {{2, 4}, {2, 4}}],
  MultiedgeStyle → True, VertexNumber → True]
```

Out[189]=



- Vozlišči 1 in 2 sta povezani z eno povezavo, zato se pojavi v drugem stolpcu prve vrstice in v prvem stolpcu druge vrstice 1.
- Vozlišči 2 in 4 sta povezani z dvema povezavama, zato je v matriki v četrtem stolpcu druge vrstice in v drugem stolpcu četrte vrstice 2.

$$\begin{array}{rc} & \begin{array}{cccc}\text{stolpec } 1 & \text{stolpec } 2 & \text{stolpec } 3 & \text{stolpec } 4 \\ \downarrow & \downarrow & \downarrow & \downarrow\end{array} \\ \begin{array}{r}\text{vrstica } 1 \rightarrow \\ \text{vrstica } 2 \rightarrow \\ \text{vrstica } 3 \rightarrow \\ \text{vrstica } 4 \rightarrow\end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 2 \\ 0 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{pmatrix}\end{array}$$

5.2. VGRAJENI GRAFI

Grafe nazorno prikažemo z njihovimi diagrami. V diagramu narišemo vozlišča kot pike ali krogce, povezave sosednjih vožlišč pa ponazorimo s črtami, pri čemer so lahko črte neravne.

Vgrajeni grafi so grafi, katerih imena so že vgrajena v prrogramu *Mathematica*. Diagrame vgrajenih grafov v *Mathematici* izrisujemo z **ShowGraph[g]**, več grafov pa z **ShowGraphArray[Table[$\mathbf{g_1}$, $\mathbf{g_2}$, …, $\mathbf{g_n}$]]**, pri čemer so g, g_1, g_2,, …,g_n grafi.

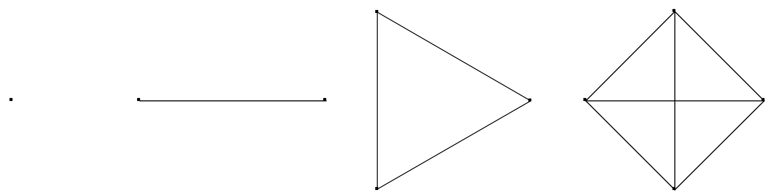
Polni graf

Graf K_n je polni graf na n vozliščih, v katerem je n vozlišč in vse možne povezave med njimi. Polne grafe navadno narišemo v obliki pravilnega mnogokotnika.

Zgled 5.2: Polni grafi na n vozliščih, kjer je n = 1, 2, 3, 4.

In[190]:= **ShowGraphArray[Table[CompleteGraph[n], {n, 1, 4}]]**

Out[190]=



Cikel

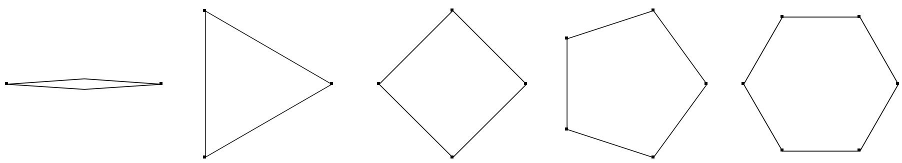
Graf C_n je cikel na n vozliščih, ki ima vozlišča $u_1, u_2, \ldots, u_n$ in povezave

$u_1u_2, u_2u_3, \ldots, u_{n-1}u_n, u_nu_1$.

Zgled 5.3: Cikli na n vozliščih, kjer je n = 2, 3, 4, 5, 6.

In[191]:= **ShowGraphArray[Table[Cycle[n], {n, 2, 6}]]**

Out[191]=



Cikla na enem vozlišču nam *Mathematica* ne izriše.

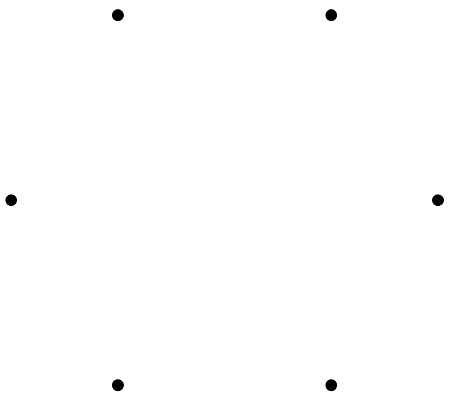
Prazni graf

Prazni graf je graf brez povezav. Prazni graf na n točkah označimo z N_n.

Zgled 5.4: Prazni graf N_6.

In[192]:=
```
ShowGraph[EmptyGraph[6]]
```

Out[192]=



Pot

Graf P_n je pot na n točkah. Graf P_n ima $n-1$ povezav in ga dobimo iz C_n z odstranitvijo katerekoli povezave.

Zgled 5.5: Pot na n vozliščih, kjer je n = 1, 2, 3, 4.

In[193]:=
```
ShowGraphArray[Table[Path[n], {n, 1, 4}]]
```

Out[193]=

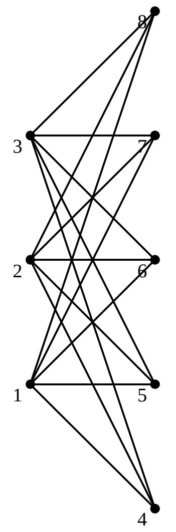
Dvodelni graf

Graf G je dvodelni, če lahko $G(V)$ razbijemo na podmnožici X in Y, tako da ima vsaka povezava grafa G eno krajišče v X in drugo v Y. Če je vsako vozlišče iz Y, govorimo o polnem dvodelnem grafu. Če je $|X| = n$ in $|Y| = m$, ga označimo z $K_{n,m}$.

Zgled 5.6: Dvodelni graf $K_{3,5}$.

```
In[194]:= ShowGraph[CompleteKPartiteGraph[3, 5],
  VertexNumber -> True]
```

Out[194]=



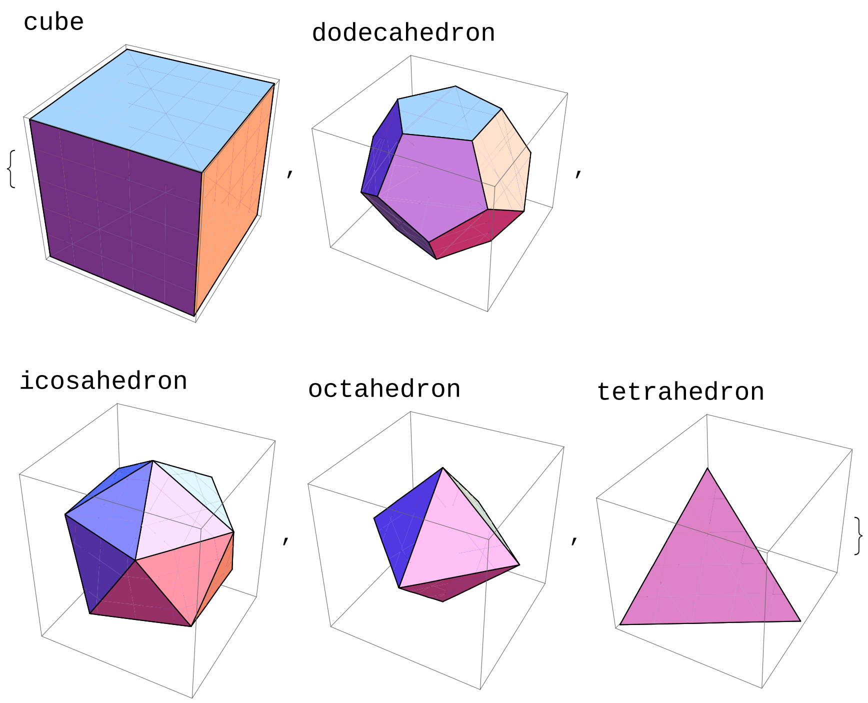
Grafi pravilnih teles (platonski grafi)

Telesa tetraeder, kocka, oktaeder, dodekaeder in ikozaeder skupaj sestavljajo peterico, ki jim pravimo Platonska telesa.

Zgled 5.7: Slike platonskih teles.

```
In[195]:= Map[Column[{Style[#[[1]], Medium], Show[#[[2]], ImageSize → 120]}] &,
  PolyhedronData["Platonic", {"Name", "Image"}]]
```

Out[195]=

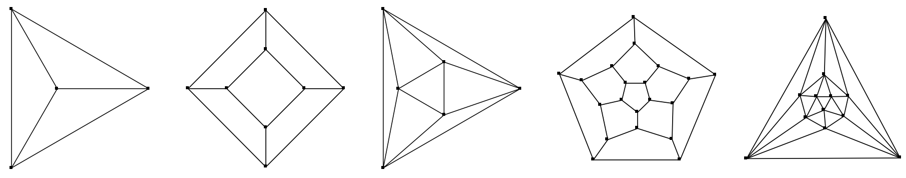


Ogrodje platonskih teles nam lahko predstavlja grafe. Oglišča teles uporabimo za vozlišča, robove pa za povezave grafa med vozlišči. Tako dobimo grafe pravilni teles ali platonske grafe. Narisani so v spodnjem zgledu.

Zgled 5.8: Grafi platonskih teles.

In[196]:=
```
ShowGraphArray[
  {TetrahedralGraph, CubicalGraph, OctahedralGraph,
   DodecahedralGraph, IcosahedralGraph}]
```

Out[196]=



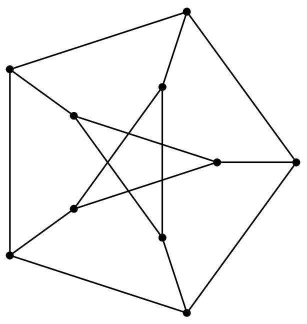
Petersenov graf

Petersenov graf je v teoriji grafov pomemben graf na desetih vozliščih z mnogimi lastnostmi. Imenuje se po danskem matematiku Juliusu Petersenu, ki ga je vpeljal leta 1892 in objavil leta 1898.

Zgled 5.9: Petersenov graf.

In[197]:=
```
ShowGraph[PetersenGraph]
```

Out[197]=

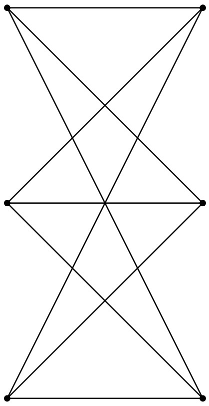


Nekatere grafe lahko dobimo na več načinov. Na primer dvodelni graf $K_{3,3}$ narišemo tudi z ukazom v spodnjem zgledu.

Zgled 5.10: Graf kletka ali 3-regularen graf z obhodom dolžine 4 in minimalmin številom vozlišč.

In[198]:=
```
ShowGraph[CageGraph[4]]
```

Out[198]=



5.3. RISANJE POLJUBNEGA GRAFA

V *Mathematici* zapisujemo povezave nekoliko drugače, in sicer bomo povezavo med vozliščema u in v pisali kot $\{u, v\}$ ali $u \rightarrow v$. Vozlišča grafa bomo označevali z zaporednimi števili 1,2,3..., n.

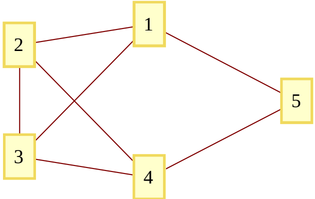
V *Mathematici* poznamo več načinov prikaza diagramov. Ogledali si jih bomo na enakem primeru.

a) Prvi način prikaza diagrama je s podajanjem povezav oblike $\{u, v\} \rightarrow m$. S številom m povemo število povezav med vozliščema. V *Mathematici* nam diagram grafa g izriše ukaz **GraphPlot[g]**. Ukaz **VertexLabeling → True** pa nam na diagramu napiše oznake vozlišč. Tako dobimo označeni graf.

Zgled 5.11:

In[199]:=
```
GraphPlot[SparseArray[{{1, 2} → 1, {2, 3} → 1, {3, 4} → 1,
    {4, 5} → 1, {5, 1} → 1, {2, 4} → 1, {1, 3} → 1}],
  VertexLabeling → True]
```

Out[199]=

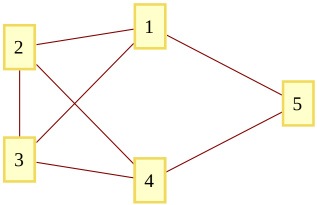


b) Povezave lahko podajamo v obliki *u*→*v*.

<u>Zgled 5.12:</u>

In[200]:=
```
GraphPlot[{1 → 2, 2 → 3, 3 → 4, 4 → 5, 5 → 1, 2 → 4, 1 → 3},
 VertexLabeling → True]
```

Out[200]=



c) Tretji način je zapis diagrama z matriko sosednosti.

<u>Zgled 5.13:</u>

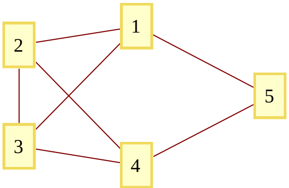
In[201]:=
```
{{0, 1, 1, 0, 1}, {1, 0, 1, 1, 0}, {1, 1, 0, 1, 0},
  {0, 1, 1, 0, 1}, {1, 0, 0, 1, 0}} // MatrixForm
```

Out[201]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

In[202]:=
```
GraphPlot[{{0, 1, 1, 0, 1},
  {1, 0, 1, 1, 0}, {1, 1, 0, 1, 0}, {0, 1, 1, 0, 1},
  {1, 0, 0, 1, 0}}, VertexLabeling → True]
```

Out[202]=

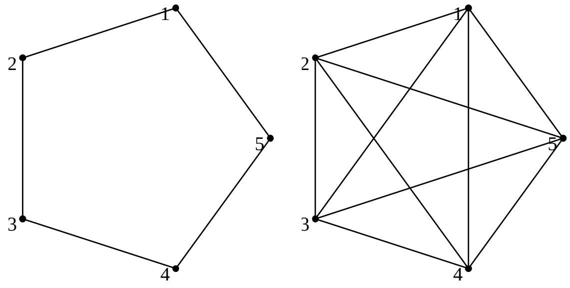


č) Nekateri grafi so že vgrajeni v paketu *Combinatorica*. Vgrajene grafe izrišemo z ukazom **ShowGraph**, medtem ko oznake vozlišč izpišemo z **VertexNumber→True**.

Zgled 5.14: Izriši cikel C_5 in polni graf K_5.

In[203]:=
```
ShowGraphArray[
  {Cycle[5], CompleteGraph[5]}, VertexNumber → True]
```

Out[203]=

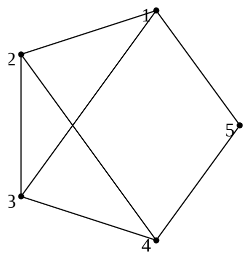


Vgrajenim grafom lahko dodajamo in odvzemamo povezave. Če prvemu grafu v zgornjem zgledu dodamo povezavi {2, 4} in {1, 3}, dobimo naš graf (zgled 5.15). Do enakega grafa pridemo, če spodnjemu grafu v zgornjem zgledu odvzamemo povezave {1, 4}, {2, 5} in {3, 5}.

Zgled 5.15:

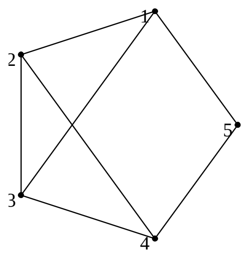
In[204]:=
```
ShowGraph[AddEdges[Cycle[5], {{2, 4}, {1, 3}}],
  VertexNumber → True]
```

Out[204]=



In[205]:=
```
ShowGraph[DeleteEdges[CompleteGraph[5],
   {{1, 4}, {2, 5}, {3, 5}}], VertexNumber → True]
```

Out[205]=

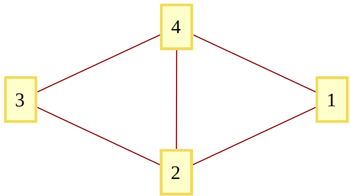


Mathematica izriše vzporedne in povratne povezave le, če ji damo ustrezen ukaz. Ukaz za izris vzporednih povezav je **MultiedgeStyle→True**, ukaz za izris povratnih povezav pa **SelfLoopStyle→True**.

Zgled 5.16: Primer uporabe ukazov *MultiedgeStyle→True* in *SelfLoopStyle→True*.

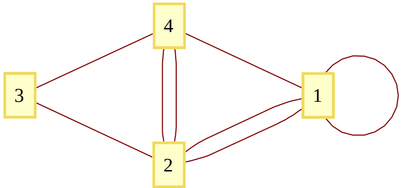
In[206]:=
```
GraphPlot[SparseArray[
  {{1, 2} → 2, {2, 3} → 1, {3, 4} → 1, {4, 1} → 1,
   {2, 4} → 2, {1, 1} → 1}], VertexLabeling → True]
```

Out[206]=



In[207]:=
```
GraphPlot[SparseArray[{{1, 2} → 2, {2, 3} → 1,
    {3, 4} → 1, {4, 1} → 1, {2, 4} → 2, {1, 1} → 1}],
 VertexLabeling → True, SelfLoopStyle → True,
 MultiedgeStyle → True]
```

Out[207]=



Vgrajene grafe včasih potrebujemo zapisane kot množico povezav ali kot sosednjo matriko. Kot množico povezav vgrajeni graf *g* zapišemo z ukazom **ToUnorderedPairs[g],** kot sosednjo matriko pa z ukazom **ToAdjacencyLists[g]**.

Zgled 5.17: Zapiši cikel C_5 kot množico povezav in sosednjo matriko.

In[208]:= **ToUnorderedPairs[Cycle[5]]**

Out[208]= {{1, 2}, {2, 3}, {3, 4}, {4, 5}, {1, 5}}

In[209]:= **ToAdjacencyLists[Cycle[5]] // MatrixForm**

Out[209]//MatrixForm=

$$\begin{pmatrix} 2 & 5 \\ 1 & 3 \\ 2 & 4 \\ 3 & 5 \\ 1 & 4 \end{pmatrix}$$

5.4. EULERJEV IN HAMILTONOV GRAF

Sprehod v grafu je Eulerjev, če vsebuje vsako povezavo na grafu natanko enkrat. Eulerjev obhod je Eulerjev sprehod, če se konča in začne v istem vozlišču. Povezani graf je Eulerjev, če vsebuje Eulerjev obhod.

Sprehod v grafu je Hamiltonov, če vsebuje vsako vozlišče na grafu natanko enkrat. Hamiltonov sprehod, ki se konča in začne v isti točki, imenujemo Hamiltonov cikel. Povezani graf je Hamiltonov, če vsebuje Hamiltonov cikel.

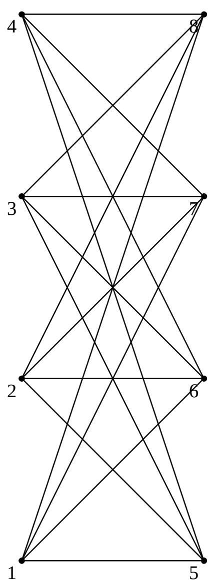
Z ustreznimi ukazi *Mathematica* za nas najde Eulerjev obhod in Hamiltonov cikel.

Ukaz **EulerianQ[g]** nam pove, ali je graf *g* Eulerjev. Če je, se nam izpiše beseda *True*, drugače pa *False*. Poljuben Eulerjev obhod grafa *g* izpišemo z **EulerianCycle[g]**.

Zgled 5.18: Preveri ali je dvodelni graf $K_{4,4}$ Eulerjev. Če je, napiši poljuben Eulerjev obhod.

```
In[210]:= ShowGraph[CompleteKPartiteGraph[4, 4],
  VertexNumber -> True]
```

Out[210]=



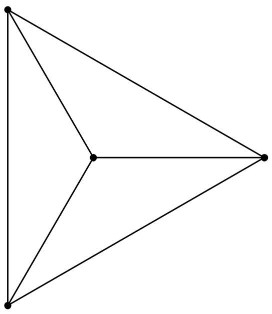

```
EulerianQ[CompleteKPartiteGraph[4, 4]]
True
EulerianCycle[CompleteKPartiteGraph[4, 4]]
{7, 2, 8, 1, 5, 4, 6, 3, 7, 4, 8, 3, 5, 2, 6, 1, 7}
```

Podobno nam ukaz **HamiltonianQ[g]** pove, ali obstaja Hamiltonov cikel v grafu *g*. Odgovor je pritrdilen ali negativen (*True* ali *False*). Poljuben Hamiltonov sprehod grafa *g* izpišemo z **EulerianPath[g]**, poljuben Hamiltonov cikel pa z ukazom **HamiltonianCycle[g]**. Vse Hamiltonove cikle grafa *g* poda ukaz **HamiltonianCycle[g, All]**. Če dodamo ukaz **Length**, lahko vse Hamiltonove cikle grafa *g* tudi preštejemo.

Zgled 5.19: Preveri in potrdi, da v grafu tetraedra obstaja Hamiltonov cikel. Zapiši poljubno Hamiltonovo pot in cikel ter ga na grafu označi. Zapiši in preštej še vse možne Hamiltonove cikle v podanem grafu.

In[211]:= **ShowGraph[TetrahedralGraph]**

Out[211]=



In[212]:= **HamiltonianQ[TetrahedralGraph]**

Out[212]= True

In[213]:= **HamiltonianPath[TetrahedralGraph]**

Out[213]= {1, 2, 3, 4}

In[214]:= **HamiltonianCycle[TetrahedralGraph]**

Out[214]= {1, 2, 3, 4, 1}

In[215]:= **HamiltonianCycle[TetrahedralGraph, All]**

Out[215]= {{1, 2, 3, 4, 1}, {1, 2, 4, 3, 1}, {1, 3, 2, 4, 1},
{1, 3, 4, 2, 1}, {1, 4, 2, 3, 1}, {1, 4, 3, 2, 1}}

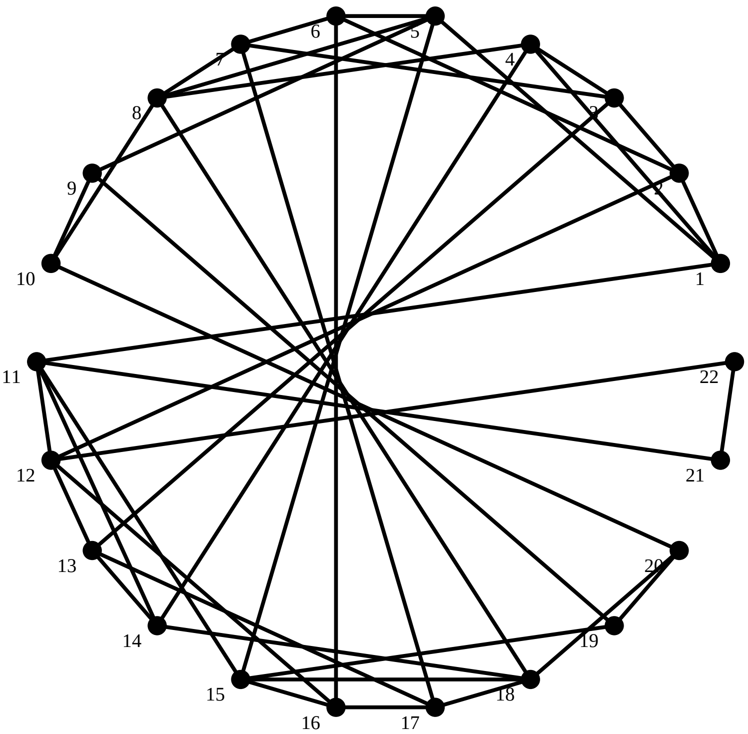
In[216]:= **Length[HamiltonianCycle[TetrahedralGraph, All]]**

Out[216]= 6

Zgled 5.20: Ugotovi ali na resonančnem grafu benzenoidnega grafa, ki izhaja iz kemije, obstaja Hamiltonov cikel. Če obstaja, najdi Hamiltonovo pot in cikel. Hamiltonov cikel na grafu označi. Prejštej vse možne cikle v podanem grafu.

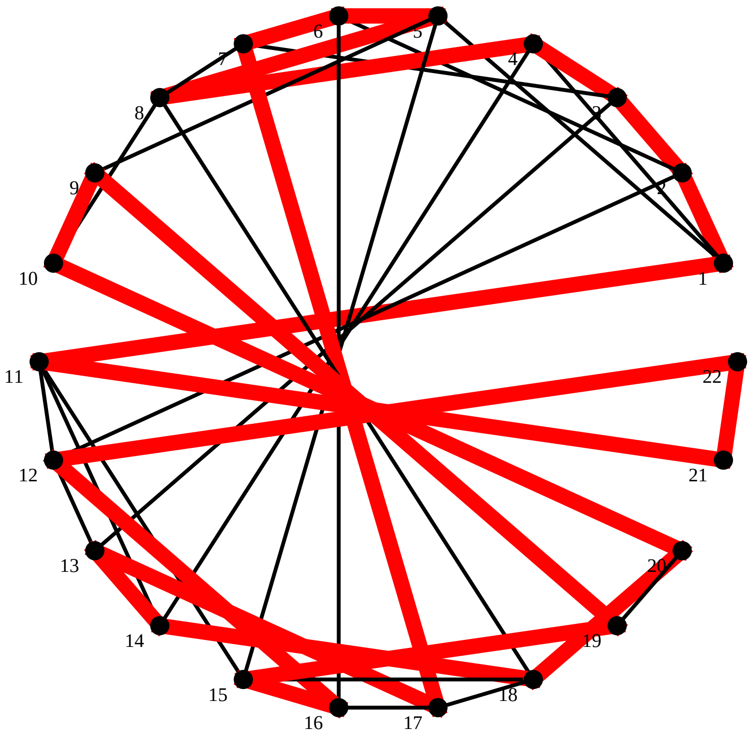
```
In[217]:= ShowGraph[gg = FromAdjacencyMatrix[
    {{0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
     {1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
     {0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0},
     {1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0},
     {1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0},
     {0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0},
     {0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0},
     {0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0},
     {0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0},
     {0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0},

     {1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0},
     {0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1},
     {0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0},
     {0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0},
     {0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0},
     {0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0},
     {0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0},
     {0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0},
     {0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0},
     {0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0},

     {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1},
     {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0}}],
  VertexNumber -> True]
```

Out[217]=



```
In[218]:= HamiltonianQ[gg]

Out[218]= True

In[219]:= HamiltonianPath[gg]

Out[219]= {1, 2, 3, 4, 8, 5, 6, 7, 17, 13, 14,
           18, 20, 10, 9, 19, 15, 16, 12, 22, 21, 11}

In[220]:= HamiltonianCycle[gg]

Out[220]= {1, 2, 3, 4, 8, 5, 6, 7, 17, 13, 14, 18,
           20, 10, 9, 19, 15, 16, 12, 22, 21, 11, 1}

In[221]:= ShowGraph[
           Highlight[gg, {Partition[HamiltonianCycle[gg], 2, 1]},
            HighlightedEdgeColors → Red], VertexNumber -> True]
```

Out[221]=




```
In[222]:= Length[HamiltonianCycle[gg, All]]

Out[222]= 3576
```

ZAKLJUČEK

S programom *Mathematica* se tekom študija matematike nisem nikoli srečala. Kot študentka matematike in fizike sem še največ o tem programu slišala pri fiziki, in sicer pri predmetu Matematične metode v fiziki. Kasneje sem pri tem predmetu morala narediti semirarsko nalogo ter si pri izračunah pomagati s programom *Mathematica*. Začetek je bil po mojih izkušnjah zahteven, še posebno, ker sem uporabljala *Mathematico* 5.0.

V diplomski nalogi je bilo delo mnogo lažje. Mogoče zaradi nekaj predhodnjih izkušenj in dobre pomoči, ki jo nudi prenovljeni program *Mathematica* 7.0. V tej novi različici namreč najdemo boljše in razširjene priročne palete. Zame najpomembnejša je bila paleta *Writing Assistant,* v kateri sem našla skoraj vse oblikovne možnosti, ki sem jih potrebovala pri pisanju moje diplomske naloge.

Prav zaradi prenovljenih palet menim, da je postal program veliko primernejši tudi za šrednješolce. Seveda pa še vedno pomembno znanje angleškega jezika, brez katerega v *Mathematici* ne gre.

LITERATURA

- S. Wolfram, *The Mathematica book*, Champaign (IL), Cambridge, Cambridge University Press, 1996
- M. L. Abell, J. P. Braselton, *Mathematica by Example*, Boston, Academic Press, 1992
- A. Fajmut, *Uporaba programskega paketa Mathematica v fiziki*, Maribor, Pedagoška fakulteta Maribor, 2002
- S. Pemmaraju, S. Skiena, *Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*, Cambridge, Cambridge University Press, 2003
- S. Klavžar, P. Žigert, *Izbrana poglavja uporabne matematike*, Maribor, Pedagoška fakulteta Maribor, 2002
- R. J. Wilson, J. J. Watkins, *Uvod v teorija grafov*, Društvo matematikov, fizikov in astronomov Slovenije, Ljubljana, 1997
- J. A. Čibej, *Matematika, Kombinatorika, Verjetnostni račun, Statistika*, Ljubljana, DZS, 1997
- http : // mathworld.wolfram.com, 17.5.2009
- http://www.cs.sunysb.edu/~skiena/combinatorica/paper.pdf, 17.5.2009