»Mladi za napredek Maribora 2013«

30. srečanje

Mathematica PRI MATEMATIKI V 1. IN 2. LETNIKU SPLOŠNEGA GIMNAZIJSKEGA PROGRAMA

Raziskovalno področje: matematika

Raziskovalna naloga

Avtor: DOMEN ULBL, JANINA ULBL

Mentor: IRENA KOSI ULBL

Ġola: II. GIMNAZIJA MARIBOR

Maribor, februar 2013

»Mladi za napredek Maribora 2013«

30. srečanje

Mathematica PRI MATEMATIKI V 1. IN 2. LETNIKU SPLOŠNEGA GIMNAZIJSKEGA PROGRAMA

Raziskovalno področje: matematika

Raziskovalna naloga

Maribor, februar 2013

KLJUČNA DOKUMENTACIJSKA INFORMACIJA

ŠD
KG računalniški program *Mathematica,* matematične naloge
AV
SA
KZ 2000 Maribor
ZA
LI 2013
IN *Mathematica* PRI MATEMATIKI V 1. IN 2. LETNIKU SPLOŠNEGA GIMNAZIJSKEGA PROGRAMA
TD Raziskovalna naloga
OP 81 strani, 9 slik, 4 tabele, 33 grafov
IJ SL
AI

Mathematica sodi med zmogljivejše računalniške programe, ki omogočajo računanje s števili, simbolično računanje in risanje grafov. Z nalogo smo želeli raziskati, če je program primeren za uporabo pri matematiki v nižjih letnikih gimnazije. Na osnovi proučevanja različnih ukazov v programu *Mathematica* smo rešili mnoge matematične naloge. V nekaterih primerih smo besedilo naloge sami razširili ali si novo nalogo izmislili in tako prikazali čim večjo uporabnost *Mathematice*. V drugem delu raziskovalne naloge smo izvedli učno uro matematike z uporabo omenjenega programa ter proučili odziv dijakov. Analiza odgovorov anketnega vprašalnika je med drugim pokazala, da je program *Mathematica* med anketiranimi dijaki razmeroma nepoznan, da je večina dijakov visoko ocenila zanimivost uporabe programa, prav tako bi večina dijakov program vključila v obravnavo snovi pri rednem pouku. Ugotovili pa smo, da *Mathematica* ne bi bistveno dvignila motivacije dijakov pri pouku matematike.

KAZALO VSEBINE

KAZALO SLIK

KAZALO TABEL

KAZALO GRAFOV

POVZETEK

Mathematica sodi med zmogljivejše računalniške programe, ki omogočajo računanje s števili, simbolično računanje in risanje grafov. Z nalogo smo želeli raziskati, če je program primeren za uporabo pri matematiki v nižjih letnikih gimnazije. Na osnovi proučevanja različnih ukazov v programu *Mathematica* smo rešili mnoge matematične naloge. V nekaterih primerih smo besedilo naloge sami razširili ali si novo nalogo izmislili in tako prikazali čim večjo uporabnost *Mathematice*. V drugem delu raziskovalne naloge smo izvedli učno uro matematike z uporabo omenjenega programa ter proučili odziv dijakov. Analiza odgovorov anketnega vprašalnika je med drugim pokazala, da je program *Mathematica* med anketiranimi dijaki razmeroma nepoznan, da je večina dijakov visoko ocenila zanimivost uporabe programa, prav tako bi večina dijakov program vključila v obravnavo snovi pri rednem pouku. Ugotovili pa smo, da *Mathematica* ne bi bistveno dvignila motivacije dijakov pri pouku matematike.

ABSTRACT

Mathematica is a powerful computer programme which provides numerical as well as symbolic calculation and drawing graphs of functions. In our research work we tried to explore if the programme is suitable for use in education in first years of gymnasium. Based on investigation of different types of programme functions we solved lots of mathematical problems. In some cases we extended given problems or made a problem ourselves to show how powerful *Mathematica* is. In the second part of our work we performed teaching mathematics using *Mathematica* and explored feedback of participating students. The analysis of the questionnaire's answers given by those students showed that *Mathematica* is not well known. Most of them gave good marks for use of the programme. Majority of the students would integrate use of *Mathematica* in regular math lessons. But it has been found out that *Mathematica* wouldn't increase motivation for learning mathematics.

ZAHVALA

Mentorici se iskreno zahvaljujeva za nasvete, pomoč in spodbudo pri nastajanju raziskovalne naloge.

1. UVOD

1.1. Namen

Spoznati želimo program *Mathematica* in se seznaniti z njegovo zgodovino, razvojem in s področji, kjer se uporablja. Raziskati želimo čim več različnih ukazov v programu *Mathematica*, ki bi nam koristili pri reševanju srednješolskih matematičnih problemov. Naučiti se želimo reševanja nekaterih nalog s programom *Mathematica* in ob tem pregledno zajeti večino matematičnih vsebin 1. in 2. letnika gimnazijskih programov. Zanima nas, kako bi se uporaba programa obnesla pri pouku in kakšen bi bil odziv dijakov.

1.2. Cilji

- Raziskati osnovne funkcije programa *Mathematica*.
- S programom *Mathematica* rešiti čim več različnih tipov nalog iz matematičnih vsebin 1. in 2. letnika.
- Izvesti praktičen prikaz uporabe programa pri uri matematike.
- Z anketo zbrati mnenja dijakov o programu *Mathematica* ter na osnovi analize odgovorov strniti ugotovitve.
- Na podlagi rezultatov ankete odgovoriti na raziskovalna vprašanja ter potrditi oz. ovreči zastavljene hipoteze.

1.3. Raziskovalna vprašanja

- Ali lahko dijak sam le ob uporabi obstoječe literature (brez uvajalnega tečaja) usvoji uporabo osnovnih ukazov v programu *Mathematica*?
- Ali lahko naloge iz večine matematičnih vsebin 1. in 2. letnika gimnazijskega programa rešimo s programom *Mathematica*?
- Ali bodo dijaki pozitivno sprejeli uporabo programa *Mathematica* pri pouku in se ga želeli naučiti uporabljati?

1.4. Hipoteze

- Ob ustreznem matematičnem znanju in zadostnem poznavanju angleškega jezika se dijak lahko sam nauči uporabe osnovnih ukazov v programu *Mathematica.*
- S programom *Mathematica* lahko rešimo mnoge naloge iz različnih matematičnih vsebin v 1. in 2. letniku gimnazijskega programa.

- Ob osnovnem predznanju uporabe programa nalogo z *Mathematico* rešimo hitreje, kakor če bi jo reševali 'peš.'
- Program bi pritegnil predvsem tiste dijake, ki so na področju matematike uspešni ali razmišljajo o študiju, povezanem z matematiko.
- Če bi imeli možnost, bi dijaki uporabo programa vključili v redni pouk.
- Uporaba programa pri pouku bi povečala motiviranost dijakov za delo.
- Dijaki nimajo interesa za uporabo programa *Mathematica* doma.
- Dijaki se niso pripravljeni naučiti osnov uporabe programa *Mathematica*.

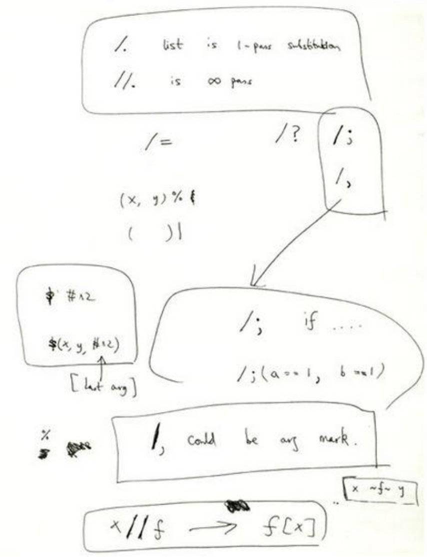
Predvidevamo, da bomo ob osvojitvi zastavljenih ciljev podrobno spoznali nekatere zmogljivosti programa *Mathematica* na področju reševanja problemov srednješolske matematike in ugotovili, ali je uporaba tega programa dovolj preprosta za dijake splošnega gimnazijskega programa. Na osnovi analize odgovorov anketiranih dijakov in na osnovi mnenja avtorjev raziskovalne naloge bomo presodili, če bi bilo program *Mathematica* primerno vključiti v redni pouk matematike.

2. O PROGRAMU *Mathematica*

2.1. Zgodovina

Z napredkom računalništva se je po letu 1960 začel razvoj programov, ki so omogočali reševanje matematičnih problemov. Bili so specializirani za posamezna področja: številčno, algebrsko, grafično in druga področja. Pri podjetju Wolfram so želeli ustvariti orodje, ki bi omogočalo raznolike funkcije in združevalo mnoga področja. Hoteli so, da bi z majhnim naborom funkcij zadostili široki paleti zahtev uporabnikov. S tem namenom so razvili nov računalniški jezik.

Prva verzija programa je bila izdana 23. junija 1988 (slika 2). Revija Business Week ga je uvrstila med deset najpomembnejših novih izdelkov tistega leta. Ožja strokovna javnost je bila nad njim navdušena, označili so ga za rezultat prave intelektualne revolucije. Pri tem je imel veliko zaslug glavni programer Stephen Wolfram. Kljub temu, da je opravil glavnino dela pri oblikovanju programa, mu ni dodelil imena. Razmišljal je o imenih Omega ali PolyMath. Svoje predloge je predstavil prijatelju Stevu Jobsu, ustanovitelju podjetja Apple. Ta mu je svetoval, naj program poimenuje *Mathematica*. Wolfram je predlog sprva zavrnil, kasneje pa spoznal, da je pravzaprav ustrezen, saj pove bistvo programa. To ime se je obdržalo.

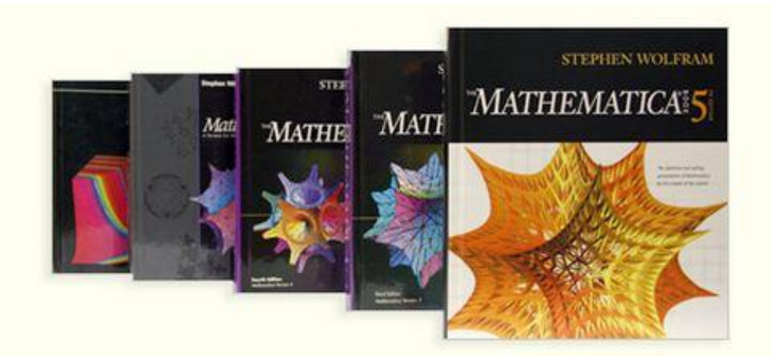


Slika 1: Osnutek programa *Mathematica* leta 1986 (vir: http://www.wolfram.com/company/scrapbook/page02.html#preWRI_InventingTheLanguage)

Slika 2: Verzija 1.0 leta 1988 na računalniku Macintosh (vir: http://www.wolfram.com/company/scrapbook/page03.html#1988_StylishMathematicaStartup)

2.2. Razvoj

Med letoma 1988 (izid *Mathematice*, verzija 1.0) in 2012 (izid najnovejše verzije 9.0.0) je program doživel mnoge spremembe – predvsem dodajanje novih funkcij, večanje zmogljivosti in prilagajanje razvoju računalniške tehnologije. Nekaj grafičnih podob posameznih verzij je prikazanih na sliki 3.



Slika 3: Verzije programa *Mathematica*
(vir: http://www.wolfram.com/company/scrapbook/page09.html#2003_MathematicaBook)

V tabeli 1 je pregled razvoja programa (nekaterih verzij); navedene so tiste značilnosti, v katerih se novejša verzija bistveno razlikuje od svoje predhodnice.

Tabela 1: Pregled verzij programa *Mathematica*

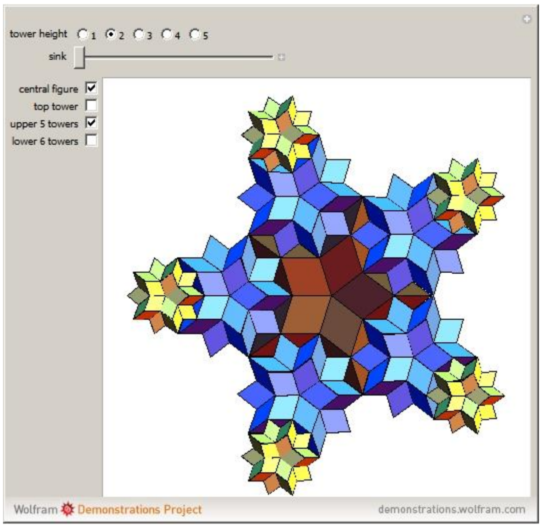
verzija	leto izida	značilnosti
1.0	1988	
1.2	1989	- zmogljive grafične funkcije - iskanje največjega skupnega delitelja - faktorizacija in iskanje inverznih funkcij
2.0	1991	- izboljšave linearne algebre - reševanje diferencialnih enačb - funkcija odkrivanja in odpravljanja napak - zvočna podpora in podpora tujih znakov
2.1	1992	- nadgradnja načrtovanja algoritma - Quick Time animacije - podpora za Windows 3.1
2.2	1993	- različica, prirejena za Linux - pomoč uporabnikom - reševanje parcialnih diferencialnih enačb in sistemov linearnih enačb
3.0	1996	- računanje z intervali - nove funkcije za poenostavljanje in preoblikovanje izrazov - nove specialne funkcije
4.0	1999	- izboljšave pri hitrosti in učinkovitosti numeričnih izračunov - objavljanje dokumentov v različnih oblikah - preverjanje črkovanja - izboljšane funkcije za analizo podatkov
4.1	2000	- zmanjšanje porabe računalniškega pomnilnika - izboljšave hitrosti pri funkcijah - zvočna podpora
4.2	2002	- izboljšano linearno programiranje - novi uvozni in izvozni programi

		- paket orodij, uporabnih pri kombinatoriki in teoriji grafov
5.0	2003	- večja hitrost izvajanja operacij - novi algoritmi za reševanje enačb in neenakosti - podpora dodatnih formatov
5.1	2004	- napredek pri vizualizaciji - izboljšana orodja za reševanje diferencialnih enačb - vgrajena povezava z bazo podatkov
6.0	2007	- izboljšanje grafike - omogočanje funkcij za nabor podatkov, uporabnih pri drugih vedah (fizika, kemija, geografija, jezikoslovje …)
7.0	2008	- integrirana obdelava slik - izboljšana statistična analiza
7.0.1	2009	- izboljšanje funkcij za obdelavo slik - podpora za Windows 7
8.0	2010	- izboljšanje risanja grafov - vgrajen spletni brskalnik
8.0.4	2011	- nove možnosti v meniju - izboljšana podpora za Linux - izboljšanje hitrosti in varnosti spletnega brskalnika
9.0	2012	- dodani so najnovejši podatki iz področij različnih znanosti - izboljšano delovanje na področju verjetnostnega računa in statistike - direktna povezava z različnimi socialnimi omrežji

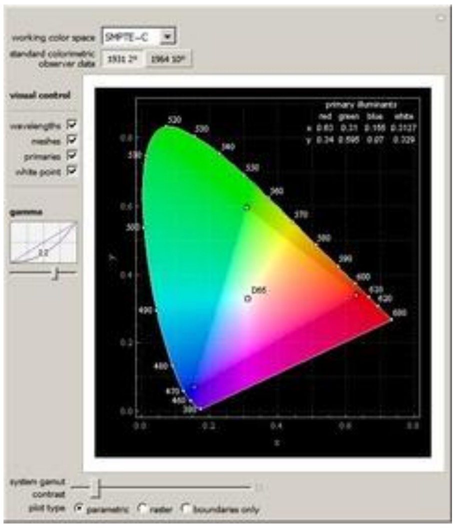
Zgodovinski pregled razvoja programa *Mathematica* smo povzeli po [9].

2.3. Uporabnost

V virih [5] in [6] zasledimo, da se je program sprva uporabljal predvsem pri matematiki, fiziki in tehnoloških vedah. Kasneje, z razvijanjem programa in dodajanjem orodij ter podatkov, se je njegova uporaba razširila in tako danes zajema mnoga področja: kemijo (slika 7), biologijo, geografijo (slika 6), medicino (slika 8), finance, arhitekturo, inženirstvo … Z *Mathematico* si strokovnjaki pomagajo pri razvijanju tehničnih produktov (slika 9), statistiki, na finančnih trgih in pri oblikovanju programske opreme.



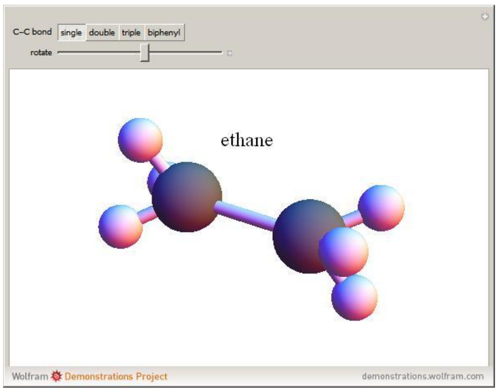
Slika 4: Trirazsežni geometrijski objekt (vir: http://demonstrations.wolfram.com/RhombicHexecontahedronTower/)



Slika 5: Diagram CIE barvnega prostora (vir: http://demonstrations.wolfram.com/CIEChromaticityDiagram/)

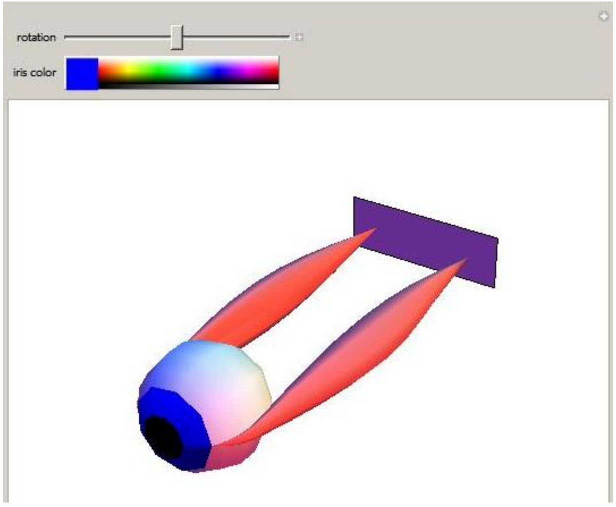
Kakor so raznolika področja uporabe programa, tako tudi uporabniki niso le manjša skupina strokovnjakov; program je razen matematikom, fizikom in inženirjem v pomoč tudi podjetnikom, oblikovalcem, biologom, kemikom, jezikoslovcem in umetnikom po vsem svetu.

Slika 6: Zemljevid angleško govorečih območij po svetu (vir: http://demonstrations.wolfram.com/CountryGroups/)

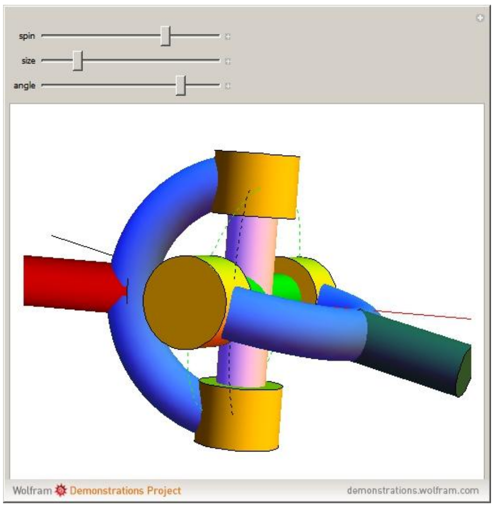


Slika 7: Prikaz molekule etana (vir: http://demonstrations.wolfram.com/RotationAboutCarbonCarbonBonds/)

Program *Mathematica* je pomembno mesto dobil tudi v šolstvu. Uporablja ga 50 največjih univerz na svetu, v poučevanje ga vključujejo profesorji, študentom ponuja nove možnosti pri učenju in razumevanju snovi.



Slika 8: Očesno zrklo z mišicama obračalkama (vir: http://demonstrations.wolfram.com/EyeballAndTwoMuscles/)



Slika 9: Mehanični zglob (vir: http://demonstrations.wolfram.com/UniversalJoint/)

Zapišemo še, da obstaja tudi blog podjetja WolframAlpha [10], v katerem so objavljene novosti v zvezi s programom *Mathematica*, kakor tudi mnogi članki, ki opisujejo uporabo programa na različnih področjih.

3. SPLOŠNA NAVODILA ZA DELO S PROGRAMOM *Mathematica*

Program *Mathematica* je zelo razširjen in sodi med najzmogljivejše programe, ki omogočajo računanje s števili, simbolično računanje ter risanje grafov, in pri tem skušajo pokriti čim več področij matematike. Ko zaženemo *Mathematico*, se v pomnilnik naložijo mnoge, toda še zdaleč ne vse funkcije, s katerimi program razpolaga. Nekatere specializirane funkcije in postopke naložimo ločeno iz datotek imenika *Mathematice* na trdem disku. Te datoteke so oblike *ime_datoteke.m*. Posamezen paket naložimo z ukazom `In:<<ImePaketa`` ali `In:Needs[»ImePaketa`«]`. Tako lahko naložimo nekatera področja matematike – na primer kombinatoriko, posebne ukaze v statistiki, dodajanje legend grafom funkcij, ki so narisani v istem koordinatnem sistemu, različne zemljevide, koledar, …

3.1. Celice

Mathematica ustvari datoteko, ki jo imenujemo beležnica (notebook). Osnovni gradnik beležnice je celica. Vsaka celica je označena z oglatim zaklepajem na desni strani beležnice. Vsebuje lahko podcelico, ta spet podpodcelico in tako naprej (govorimo o tako imenovani drevesni strukturi oziroma gnezdenju). Celico označimo z enim klikom na zaklepaj, z dvema klikoma pa jo zapremo (če je odprta), oziroma odpremo (če je zaprta). Celico lahko razdelimo tako, da v meniju kliknemo *Cell→Divide Cell*. Na mestu kurzorja se bo celica razdelila na dva dela. Seveda lahko naredimo tudi obraten postopek združevanja celic. Označimo zaklepaje celic, ki jih želimo združiti, in v meniju kliknemo *Cell→Merge Cells*. Obstaja več vrst celic: vhodne, izhodne, naslovne, tekstovne, grafične, … Beležnico lahko oblikujemo kot neke vrste elektronsko knjigo s poglavji, podpoglavji, razdelki, …, ki vsebujejo besedilo, računske primere, grafične prikaze. Vrsto celice lahko izberemo s klikom v meniju na *Format→Style*. Z namenom, da beležnica ni predolga, lahko posamezne celice (z vgnezdenimi podcelicami) zapremo z dvoklikom na celični zaklepaj. Ponovni dvoklik na ta zaklepaj celico spet odpre. Nove izraze oziroma novo besedilo običajno vnašamo na koncu beležnice, seveda pa lahko spreminjamo tudi že obstoječe celice. *Mathematica* vsak vnos (vhodno vrstico) števil či - oznaka `In[]`, z isto številko je označena tudi izhodna vrstica - oznaka `Out[]`. Pri tem upošteva časovno zaporedje vnosov. Če spreminjamo katero izmed prejšnjih celic, se časovno zaporedje vnosov, ki ga odražajo številke, ne ujema več s prostorskim.

Pri vnašanju besedila si lahko pomagamo s paletama »Basic Math Assistant«, ki ima vgrajenih veliko matematičnih funkcij in omogoča enostavno vnašanje najrazličnejših matematičnih izrazov in »Writing Assistant«, ki služi kot preprost urejevalnik besedila.

3.2. Osnovna pravila pri delu z *Mathematico*

Program *Mathematica* je interaktiven. Ko uporabnik vnese izraz in pritisne tipki SHIFT-ENTER, *Mathematica* vrne vrednost izraza. Pritisk na tipko ENTER, kot je to običajno, če želimo dobiti rezultat, le prelomi vrstico znotraj celice.

Pri zapisovanju matematičnih izrazov pogosto uporabljamo oklepaje. V *Mathematici* imajo različne oblike oklepajev posebne funkcije:

- (): Z okroglimi oklepaji grupiramo elemente in na tak način na primer spreminjamo »naravni vrstni red« računskih operacij.
- []: Z oglatimi oklepaji obdamo argumente funkcije ali ukaza.
- {}: V zavitih oklepajih naštejemo elemente seznama.
- [[]]: V dvojnih oglatih oklepajih navedemo indeks elementa seznama.
- (**): Med okrogle oklepaje z zvezdico zapišemo komentar.

Pri zapisovanju izrazov z oklepaji so le-ti vijolično-rdečkaste barve, dokler jih ne zaključimo z zaklepajem. Takrat spremenijo barvo v črno. Barvne oznake oklepajev so nam v pomoč pri vnašanju izrazov z več oklepaji, saj nas program opozarja, da vseh oklepajev še nismo zaključili z zaklepaji. Več praktične uporabe različnih vrst oklepajev bomo prikazali v nadaljevanju ob različnih primerih.

<u>Opomba:</u> V nadaljevanju bomo pri vseh zgledih in reševanju nalog z *Mathematico* vhodne celice označili z modro, izhodne celice pa z oranžno barvo. Oznake `In[]` in `Out[]` na levi strani celice bomo, razen v prvem zgledu, izpustili.

<u>Zgled:</u> Tvorili bomo seznam »teden« in poiskali 5. element v tem seznamu.

```
In[1]:= teden = {ponedeljek, torek, sreda, četrtek, petek, sobota, nedelja}
        (*definirali smo seznam "teden" s 7 elementi*)

Out[1]= {ponedeljek, torek, sreda, četrtek, petek, sobota, nedelja}

In[2]:= teden[[5]]

Out[2]= petek
```

Pri zapisovanju ukazov v *Mathematici* lahko uporabljamo rezultate, ki smo jih predhodno pridobili. Tako simbol `%` pomeni prejšnji rezultat, simbol `%%` predprejšnji rezultat in tako naprej. Simbol `%n` pomeni isto kot `Out[n]`.

<u>Zgled:</u>

```
2 + 2        2 + %        % - %%

4      →     6      →     2
```

V tem razdelku omenimo še naslednje posebnosti *Mathematice*:

- imena konstant pišemo z veliko začetnico – na primer Ludolfovo število, Eulerjevo število in imaginarno enoto (π, e in i) bomo vnesli kot `Pi`, `E` in `I`
- argumente funkcij bomo obdali z oglatimi oklepaji – torej `f[x]` in ne `f(x)` kot običajno

- imena vgrajenih funkcij se začnejo z veliko začetnico, na primer `Simplify`; če je ime sestavljeno iz dveh besed, sta obe zapisani z veliko začetnico in skupaj, na primer `ParametricPlot`
- znak \ na koncu skupine števk pomeni, da se število nadaljuje v naslednji vrstici

V program *Mathematica* je vgrajena tudi obsežna pomoč. Do nje dostopamo na več načinov. Ena izmed možnosti je preko menija *Help→Documentation Center*, ali pa uporabimo virtualni priročnik *Virtual Book*. V obeh primerih so opisane zmožnosti *Mathematice* po poglavjih in podpoglavjih, pri čemer najdemo opise funkcij posameznih ukazov in mnoge primere. Če želimo pomoč za posamezni ukaz, pred njega zapišemo ? in pritisnemo tipki SHIFT-ENTER ali pa ukaz enostavno označimo in pritisnemo tipko F1:

- `?Expand` + SHIFT-ENTER
- `Expand` + F1

V drugem primeru dobimo poleg izčrpne pomoči še informacije o sorodnih ukazih.

3.3. Operacije in relacije

Za operacije seštevanje, odštevanje, množenje, deljenje in potenciranje uporabimo simbole +, -, *, / in ^. Znak * lahko nadomestimo tudi s presledkom med faktorjema. *Mathematica* upošteva vrstni red operacij, kot ga poznamo v matematiki. Če imamo operacije enake prednosti, uporabi naravni vrstni red, torej izvaja operacije od leve proti desni. Vrstni red operacij lahko spremenimo z uporabo okroglih oklepajev. Osnovne računske operacije si oglejmo na nekaterih zgledih.

<u>Zgledi:</u>

Izračunajmo vrednost izraza z različnimi operacijami (kvadriranje ima prednost pred množenjem, to pa pred seštevanjem).

```
-8 + 6 * 4 ^ 2
```

```
88
```

V izraz postavimo oklepaje – seštevamo pred množenjem.

```
(-8 + 6) * 4 ^ 2
```

```
-32
```

Če dodamo še ene oklepaje, opravimo najprej seštevanje, nato množenje in nazadnje kvadriranje.

```
((-8 + 6) * 4) ^ 2
```

```
64
```

Rezultati se v vseh treh primerih seveda razlikujejo.

Pri pisanju decimalnih številk uporabljamo decimalno piko in ne vejice. Rezultat bo zaokrožen na 6 mest, kot je v programu privzeto.

Oglejmo si deljenje. *Mathematica* računa natančno, zato dobimo

```
10 / 6
```

$$\frac{5}{3}$$

Če želimo rezultat, zapisan z decimalno številko, moramo vsaj števcu ali imenovalcu dodati piko.

```
10. / 6
```

```
1.66667
```

Zaokroženi približek rezultata dobimo tudi s funkcijo `N`. Pri tem uporabimo oglati oklepaj, v katerega zapišemo argument funkcije.

```
N[10 / 6]
```

```
1.66667
```

Enak učinek dosežemo tudi z naslednjo obliko ukaza.

```
10 / 6 // N
```

```
1.66667
```

Število mest zapisa približka rezultata lahko sami določimo tako, da pri funkciji `N` za argumentom zapišemo vejico in število želenih mest:

```
N[10 / 6, 9]
```

```
1.66666667
```

Oglejmo si še simbole za nekatere relacije. *Mathematica* uporablja simbole ==, !=, <, <=, >, >= za relacije »enak«, »različen«, »manjši«, »manjši ali enak«, »večji«, »večji ali enak«. Izraz `aRb`, kjer je `R` ena izmed naštetih relacij, ima vrednost `True`, `False` ali pa *Mathematica* vrne izraz, če ne more ugotoviti, ali izraz velja ali ne. Tako je na primer

```
10 ^ -3 < 10 ^ -4
```

```
False
```

3.4. Spremenljivke

Mathematica lahko računa s simboli (spremenljivkami), ne le s števili. Tako je

```
2 b - 5 b
```

```
-3 b
```

Posamezni spremenljivki lahko določimo vrednost z operatorjem =, na primer `b = x`. Pri tem je `b` spremenljivka, `x` pa poljuben izraz. Izjema je le primer, ko izraz v času prirejanja še nima definirane vrednosti – to pomeni, da izraz `x` ne sme vsebovati `b`, saj bi sicer dobili neskončno zanko. Od zdaj naprej bo *Mathematica* vsako spremenljivko `b` v kateremkoli izrazu, v katerem nastopa, nadomestila z izrazom `x`.

<u>Zgled:</u>

```
b = 7 (*b-ju smo priredili vrednost 7*)
```

```
7
```

→

```
2 b - 5 b
```

```
-21
```

Spremenljivki `b` lahko spet odvzamemo vrednost z ukazoma `Clear[b]` ali `b = .` Nadaljujmo prejšnji zgled:

```
b =. (*ali Clear[b]*)
```

→

```
b*b
```

```
b^2
```

3.5. Transformacijsko pravilo

Transformacijsko pravilo imenujemo izraz oblike `x→b`. To pravilo uporabimo na izrazu `y` tako, da zapišemo `y/. x→b`. Rezultat takšnega ukaza je izraz `y`, v katerem bodo vsi podizrazi oblike `x` nadomeščeni z izrazom `b`. *Mathematica* omogoča, da na danem izrazu uporabimo več transformacijskih pravil hkrati. V takem primeru zapišemo transformacijska pravila v obliki seznama.

Zgled:

```
-2 x + 4 y + 6 /. {x → -1, y → 3}
```

```
20
```

3.6. Reševanje enačb

Enačbe rešujemo z vgrajeno funkcijo `Solve`. Temu ukazu sledijo oglati oklepaji, v katere zapišemo enačbo. Pri tem pazimo, da namesto enačaja zapišemo ==.

Zgled:

```
Solve[6 x - 4 == -6 x + 8]
```

```
{{x → 1}}
```

Če rešujemo enačbo oblike $cx = b$, dobimo z ukazom `Solve` rešitev $x = \frac{b}{c}$. V primeru, ko je $b = c = 0$, ima enačba neskončno mnogo rešitev. Če želimo poiskati vse rešitve, uporabimo ukaz `Reduce`. V primeru, ko je enačba identiteta, vrne ukaz `Reduce` vrednost `True`, če je enačba protislovje, pa vrednost `False`. Pri podajanju rešitev z ukazom `Reduce` zasledimo simbola `&&` (logični in) in `||` (logični ali).

Zgled:

```
Solve[c x == b, x]
```

$$\left\{\left\{x \to \frac{b}{c}\right\}\right\}$$

```
Reduce[c x == b, x]
```

$$(c == 0 \&\& b == 0) \,||\, \left(c \neq 0 \&\& x == \frac{b}{c}\right)$$

V primeru, ko sta *b* in *c* hkrati enaka 0, predstavljajo rešitev enačbe vsa realna števila. Tega program posebej ne izpiše.

Z ukazom `Solve` rešujemo tudi sisteme enačb. V tem primeru ima funkcija `Solve` dva argumenta. Prvi je seznam enačb, drugi pa seznam neznank. Če imamo enako število enačb in neznank, lahko drugi seznam izpustimo. Obvezno pa ga moramo navesti v primeru, ko je neznank več kot enačb. Kot rezultat dobimo seznam rešitev v obliki transformacijskih pravil za neznanke.

Zgled:

```
Solve[{3 x - y + z == 1, x + y - 2 z == -9, 2 x - 2 y + z == 2}]
```

```
{{x → -1, y → 0, z → 4}}
```

Zgled:

```
Solve[{x + 3 y + 3 z == 2, x - 3 y - 3 z == 2}, {x, y}]
```

```
{{x → 2, y → -z}}
```

3.7. Grafi funkcij

Mathematica ima vgrajene različne ukaze za dvorazsežno in trirazsežno grafiko. Nekateri ukazi so zelo preprosti in rezultati upoštevajo privzete vrednosti. V tem razdelku bomo predstavili ukaz `Plot`, ki ga uporabljamo za risanje grafov funkcij ene spremenljivke (dvorazsežna grafika) in nekaj njegovih opcij.

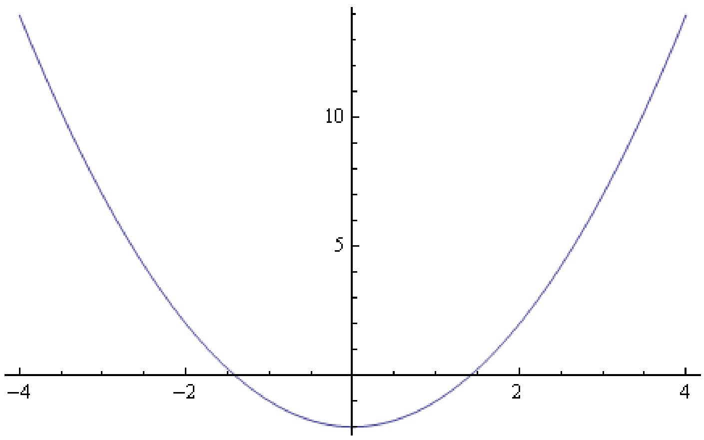
- Ukaz `Plot[f[x],{x,xmin,xmax}]` nariše dvorazsežni graf funkcije f na intervalu $x_{min} \leq x \leq x_{max}$.
- Ukaz `Plot[{f[x],g[x]},{x,xmin,xmax}]` nariše dvorazsežna grafa funkcij f in g na intervalu $x_{min} \leq x \leq x_{max}$ v isti koordinatni sistem.

Zadnji ukaz na naraven način razširimo na tri ali več funkcij.

<u>Opomba:</u> Pri predstavljanju grafov funkcij in drugih grafičnih prikazov zaradi boljše vidljivosti pri »output-u« *Mathematice* ne bomo uporabili oranžnega ozadja.

<u>Zgled</u>: Najprej bomo narisali parabolo z enačbo $y = x^2 - 2$ na intervalu $[-4,4]$.

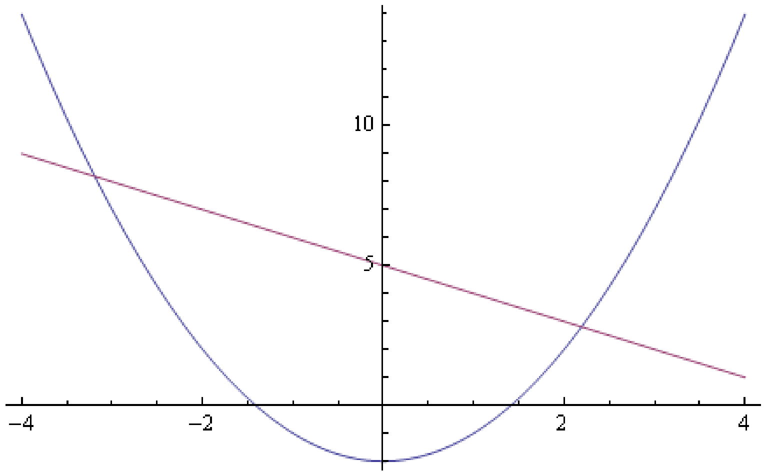
```
Plot[x^2 - 2, {x, -4, 4}]
```



Graf 1: Parabola z enačbo $y = x^2 - 2$

V drugem delu zgleda bomo paraboli v isti koordinatni sistem dodali še premico z enačbo $y = -x + 5$.

```
Plot[{x^2 - 2, -x + 5}, {x, -4, 4}]
```



Graf 2: Parabola z enačbo $y = x^2 - 2$ in premica z enačbo $y = -x + 5$

Če z izgledom grafa nismo zadovoljni, lahko ukazu `Plot` dodamo še druge argumente, ki jih imenujemo *opcije*. Opcija ima obliko `ime opcije -> vrednost opcije`. Z ustreznimi opcijami lahko označimo koordinatne osi, zapišemo ime slike (grafa), določimo barvo in debelino krivulje, določimo, naj bosta enoti na obeh koordinatnih oseh enaki, izberemo, kateri del grafa naj bo prikazan, …

Nekaj primerov opcij in njihovi pomeni:

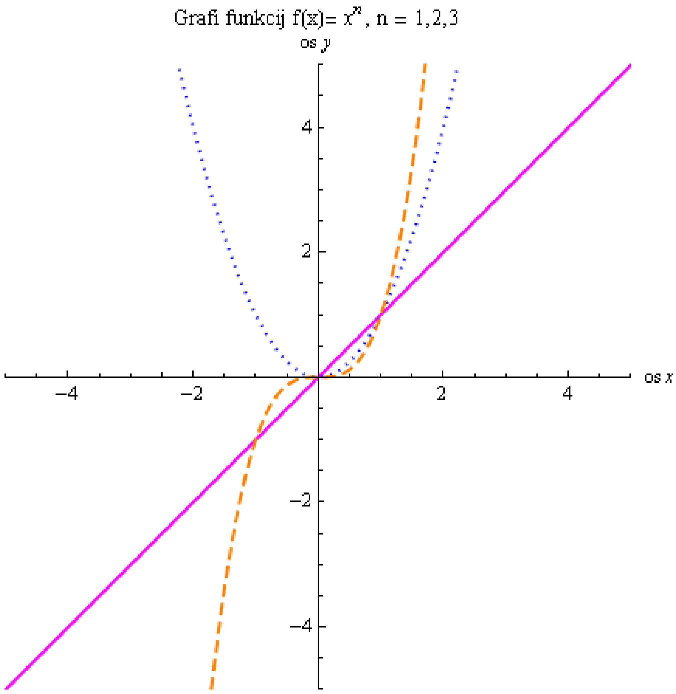
- `AspectRatio -> Automatic`: enoti na abscisni in ordinatni osi naj bosta enako veliki
- `PlotRange -> {ymin,ymax}`: prikazan naj bo del grafa, za katerega je $y_{min} \leq y \leq y_{max}$
- `AxesLabel -> {osx, osy}`: poimenovanje koordinatnih osi
- `PlotStyle -> {Dashing[{v1,v2}],Color,Thickness[v3]}`: določitev oblike (polna, črtkana, pikčasta, …), barve in debeline črte
- `PlotLabel -> :«ime slike«`: poimenovanje slike

<u>Zgled</u>: V isti koordinatni sistem bomo narisali grafe funkcij s predpisi $f(x) = x$, $g(x) = x^2$ in $h(x) = x^3$. Pri tem bomo uporabili nekaj zgoraj navedenih opcij. Ker bomo za vsako krivuljo izbrali posebne lastnosti, bomo najprej definirali funkcije *f*, g in h, nato bomo določili lastnosti posameznega grafa in na koncu z ukazom `Show` narisali vse tri grafe v isti koordinatni sistem.

Zapišimo še, da predpis za funkcijo *f* ene spremenljivke *x* definiramo z ukazom `f[x_]:=` …

```
f[x_] := x;
g[x_] := x^2;
h[x_] := x^3;
```

```
g1 = Plot[f[x], {x, -5, 5}, AxesLabel → {os x, os y} , AspectRatio → Automatic,
   PlotRange → {{-5, 5}, {-5, 5}}, PlotStyle → {Magenta, Thickness[.005]}];
g2 = Plot[g[x], {x, -5, 5}, AxesLabel → {os x, os y} , AspectRatio → Automatic,
   PlotRange → {{-5, 5}, {-1, 8}}, PlotStyle → {Blue, Dotted, Thickness[.005]}];
g3 = Plot[h[x], {x, -5, 5}, AxesLabel → {os x, os y} , AspectRatio → Automatic,
   PlotRange → {{-5, 5}, {-5, 8}},
   PlotStyle → {Orange, Dashing[{0.02, 0.01}], Thickness[.005]}];
Show[g1, g2, g3, PlotLabel → "Grafi funkcij f(x)= x^n, n = 1,2,3"]
```



Graf 3: Grafi funkcij f, g, h s predpisi $f(x) = x$, $g(x) = x^2$ in $h(x) = x^3$

3.8. Zanke v *Mathematici*

V *Mathematici* so na razpolago tudi zančne funkcije (`Do`, `While`, `For`) in ukaz `If`, ki ga pogosto uporabljamo v zvezi z zančnimi funkcijami. Predstavili bomo naslednja ukaza:

- ukaz `Do[izraz,{i,imin,imax,prirastek_i}]` ovrednoti *izraz*, pri čemer spreminja *i* od i_{min} do i_{max} s korakom *prirastek_i*
- ukaz `If[pogoj,resnica,neresnica]` ovrednoti *pogoj* in izvrši *resnico*, če je *pogoj* resničen, in *neresnico*, če je *pogoj* neresničen

Zgled: Izračunajmo vsoto vseh lihih števil od 1 do 99.

```
vsota = 0;
Do[vsota = vsota + i, {i, 1, 99, 2}]
vsota
```

```
2500
```

Zgled za funkcijo (ukaz) `If` bomo prikazali v poglavju o praštevilih.

4. MATEMATIKA V 1. IN 2. LETNIKU

V tem poglavju bomo predstavili reševanje nekaterih srednješolskih matematičnih problemov z *Mathematico*. Pri tem večino problemov predstavljajo naloge iz matematičnih učbenikov za 1. in 2. letnik splošnega gimnazijskega programa (vira [3] in [4]). Besedilo posamezne naloge bomo zapisali, dodali ustrezne komentarje v zvezi z reševanjem naloge ali s predstavitvijo programskih ukazov, nato pa navedli ustrezen ukaz (input) in rezultat, ki ga vrne *Mathematica* (output). Naloge so združene v različna matematična področja. Zapišimo še naslednji opombi:

1. Nekateri ukazi (na primer za poenostavljanje različnih izrazov, reševanje različnih enačb in neenačb, risanje grafov funkcij, …) so enaki, zato v predstavitev nismo vključevali nalog, ki sicer sodijo v različna področja, vendar pri njihovem reševanju z *Mathematico* uporabljamo iste ukaze (na primer poenostavljanje izrazov s potencami s celimi eksponenti ali s potencami z racionalnimi eksponenti).
2. V raziskovalno nalogo smo vključili matematična področja, ki smo jih pri rednem pouku obravnavali do zaključka raziskovalne naloge (snov 1. letnika in polovice 2. letnika).

4.1. Osnove teorije množic

Naloga ([3], str. 72, nal. 259): Iz univerzuma $\mathcal{U} = \{1, 2, \dots, 9\}$ zgradimo množice: $\mathcal{A} = \{1, 2, 3, 4\}$, $\mathcal{B} = \{2, 4, 6, 8\}$, $\mathcal{C} = \{7, 9\}$. Izračunajte in zapišite: $\mathcal{A} \cap \mathcal{B}$, $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cap \mathcal{C}$, $\mathcal{B} \cup \mathcal{C}$, $\mathcal{C} \cap B^{C}$, $(\mathcal{B} \cap \mathcal{C})^{C}$.

Najprej definiramo vse množice. Ker je v *Mathematici C* »zaščiten simbol« (`protected symbol`), množico $\mathcal{C}$ označimo s $\mathcal{C}1$.

```
U = {1, 2, 3, 4, 5, 6, 7, 8, 9}
```

```
{1, 2, 3, 4, 5, 6, 7, 8, 9}
```

```
A = {1, 2, 3, 4}
{1, 2, 3, 4}
```

```
B = { 2, 4, 6, 8}
{2, 4, 6, 8}
```

```
C1 = {7, 9}
{7, 9}
```

Presek množic dobimo z ukazom `Intersection` in unijo z ukazom `Union`. V oglatih oklepajih navedemo, za katere množice naj ukaz velja.

```
Intersection[A, B]
{2, 4}
```

```
Union[A, B]
{1, 2, 3, 4, 6, 8}
```

```
Intersection[A, C1]
{}
```

```
Union[B, C1]
{2, 4, 6, 7, 8, 9}
```

Množico $\mathcal{B}K$ definiramo kot komplement množice $\mathcal{B}$ glede na $\mathcal{U}$. Poiščimo elemente preseka množic $\mathcal{C}1$ in $\mathcal{B}K$ ter elemente komplementa preseka množic $\mathcal{B}$ in $\mathcal{C}1$.

```
BK = Complement[U, B]
{1, 3, 5, 7, 9}
```

```
Intersection[C1, BK]
{7, 9}
```

```
Complement[U, Intersection[B, C1]]
{1, 2, 3, 4, 5, 6, 7, 8, 9}
```

Z namenom, da predstavimo še eno računsko operacijo z množicami, bomo našo nalogo nekoliko razširili. Želimo izračunati še kartezični produkt množic $\mathcal{B}$ in $\mathcal{C}1$. Ustrezen ukaz je `CartesianProduct[B,C1]`, ki vrne pare števil v obliki seznama. Za uporabo tega ukaza moramo predhodno naložiti paket 'Kombinatorika'

```
Needs["Combinatorica`"]
```

```
CartesianProduct[B, C1]
```

```
{{2, 7}, {2, 9}, {4, 7}, {4, 9}, {6, 7}, {6, 9}, {8, 7}, {8, 9}}
```

4.2. Naravna in cela števila

4.2.1. Računanje z naravnimi in celimi števili

Naloga ([3], str. 23, nal. 76): Števila uredite po velikosti od manjšega proti večjemu.
$(-3)^2, -2^4, (-1)^8, (-6)^3, 3^2, (-3)^3, (-2)^6, -2^6$.

V *Mathematici* lahko števila uredimo po velikosti, vendar jih program izračuna (ne pusti jih v prvotni obliki). Zato števila najprej izračunamo in jih šele nato uredimo po velikosti.

```
{(-3)^2, -2^4, (-1)^8, (-6)^3, 3^2, (-3)^3, (-2)^6, -2^6}
```

```
{9, -16, 1, -216, 9, -27, 64, -64}
```

```
Sort[{(-3)^2, -2^4, (-1)^8, (-6)^3, 3^2, (-3)^3, (-2)^6, -2^6}]
```

```
{-216, -64, -27, -16, 1, 9, 9, 64}
```

Z *Mathematico* lahko na preprost način zapisujemo števila v sistemih z različnimi osnovami. Pri tem uporabimo ukaz `BaseForm[število,n]`, ki število, zapisano v desetiškem sistemu, zapiše v sistemu z bazo n. Največja baza je 36, za zapis števil v dvomestnih bazah pa uporabljamo črke na običajen način ($a = 10$, $b = 11$, $c = 12$, …).

Zgled: Zapišimo število 45 v sistemih z osnovo 3 in 16 ter število 45,67 v sistemu z osnovo 8.

`BaseForm[45, 3]`	`BaseForm[45, 16]`	`BaseForm[45.67, 8]`
1200_3	$2d_{16}$	55.52703_8

Oglejmo si še pretvarjanje zapisa števil v nedesetiškem sistemu v desetiški sistem. Število v nedesetiškem sistemu zapišemo v obliki B^^število, kjer je *B* nedesetiška baza. S pritiskom na tipki SHIFT + ENTER dobimo ustrezno število v desetiškem sistemu.

<u>Zgled:</u> Zapišimo števili 2345_6 in $bf78a_{16}$ v desetiškem sistemu.

```
6^^2345
```

```
569
```

```
16^^bf78a
```

```
784266
```

4.2.2. Algebrski izrazi

Naloga ([1], str. 29, nal. 116j): Izpostavite skupni faktor: $-x^5y^2 + x^4y^3 - 2x^3y^4$

Uporabimo ukaz `Factor`.

```
Factor[-x^5 y^2 + x^4 y^3 - 2 x^3 y^4]
```

$-x^3\, y^2\, (x^2 - x\, y + 2\, y^2)$

Naloga ([3], str. 31, nal. 146): Dan je izraz $(x-6)^2 - 3(x+2)(x-5) + 2(x^2-9)$.

a) Poenostavite izraz.

b) Rezultat razstavite.

c) Za kateri x je vrednost danega izraza 0?

Izraz poenostavimo z ukazom `Simplify`, z ukazom `Factor` izraz faktoriziramo in z ukazom `Solve` rešimo enačbo $(x-6)^2 - 3(x+2)(x-5) + 2(x^2-9) = 0$ ter tako ugotovimo, kateri x ustreza pogoju.

```
Simplify[(x - 6)^2 - 3 (x + 2) (x - 5) + 2 (x^2 - 9)]
```

```
48 - 3 x
```

```
Factor[%]
```

```
-3 (-16 + x)
```

```
Solve[-3 (-16 + x) == 0]
```

```
{{x → 16}}
```

4.2.3. Deljivost in praštevila

Naloga ([3], str. 39, nal. 156 c): Pokažite, da velja: $11 \mid (2^{33} + 3^{11})$.

Uporabimo ukaz `Divisible` in najprej navedemo izraz, katerega deljivost preverjamo, nato število, s katerim delimo. Izhodni podatek `True` nam pove, da je izraz deljiv z danim številom.

```
Divisible[2^33 + 3^11, 11]
```

```
True
```

Naloga ([3], str. 48, nal. 190 d): Zapišite vse delitelje števila 282.

Z ukazom `Divisors` poiščemo vse delitelje danega števila. Program jih zapiše v obliki seznama.

```
Divisors[282]
```

```
{1, 2, 3, 6, 47, 94, 141, 282}
```

Naloga ([3], str. 57, nal. 228 č): Dana števila zapišite kot produkt praštevil in za posamezno trojico števil poiščite največji skupni delitelj in najmanjši skupni večkratnik: $450, 1155, 1470$.

Pri razcepu števila na prafaktorje uporabimo ukaz `FactorInteger[n]`, ki vrne seznam parov `{praštevilo,eksponent}`.

```
FactorInteger[450]
FactorInteger[1155]
FactorInteger[1470]
```

```
{{2, 1}, {3, 2}, {5, 2}}
```

```
{{3, 1}, {5, 1}, {7, 1}, {11, 1}}
```

```
{{2, 1}, {3, 1}, {5, 1}, {7, 2}}
```

Za iskanje največjega skupnega delitelja uporabimo ukaz `GCD (Greatest Common Divisor)`, za najmanjši skupni večkratnik pa `LCM (Least Common Multiple)` dane trojice števil.

```
GCD[450, 1155, 1470]
```
```
15
```

```
LCM[450, 1155, 1470]
```
```
242 550
```

V zvezi s praštevili so večkrat uporabni tudi naslednji ukazi:

- `PrimeQ[n]`, ki vrne vrednost `True`, če je *n* praštevilo, oziroma `False`, če ni
- `PrimePi[n]`, ki vrne število praštevil, manjših od *n*
- `Prime[n]`, ki vrne *n*-to praštevilo

Zapisane ukaze si oglejmo na nekaterih primerih.

```
PrimeQ[{79, 80}]
```
```
{True, False}
```

```
PrimePi[20]
```
```
8
```

```
Prime[8]
```
```
19
```

Naloga ([3], str. 58, nal. 230 b): Zapišite največji skupni delitelj in najmanjši skupni večkratnik izrazov $8x + 2y$, $32x^4 - 2\,x^2y^2$, $128x^4 + \;2xy^3$.

Uporabimo ista ukaza kot v prejšnji nalogi `(GCD` in `LCM)`, dopišemo še `Polynomial`, kar označuje, da gre za izraze in ne za števila. Dobljena izraza lahko tudi faktoriziramo.

```
PolynomialGCD[8*x+2*y, 32*x^4-2*x^2*y^2, 128*x^4+2*x*y^3]
```
```
8 x + 2 y
```
```
Simplify[8 x + 2 y]
```
```
2 (4 x + y)
```
```
PolynomialLCM[8*x+2*y, 32*x^4-2*x^2*y^2, 128*x^4+2*x*y^3]
```
$$\left(4\,x^2 - x\,y\right)\,\left(128\,x^4 + 2\,x\,y^3\right)$$

```
Factor[%]
```

$2\,x^2\,(4\,x-y)\,(4\,x+y)\,\left(16\,x^2-4\,x\,y+y^2\right)$

Ob koncu tega podpoglavja bomo predstavili še zgled za uporabo funkcije If.

Zgled: Med prvimi petnajstimi naravnimi števili izpišimo posebej praštevila in posebej ostala števila.

```
Do[If[PrimeQ[j], Print[j], Print["    ", j]], {j, 1, 15}]
```

```
    1
2
3
    4
5
    6
7
    8
    9
    10
11
    12
13
    14
    15
```

4.3. Racionalna števila

4.3.1. Ulomki in potence s celimi eksponenti

Naloga ([3], str. 86, nal. 303 b): Okrajšajte ulomek: $\frac{2257}{3294}$.

Za krajšanje ulomkov, ki imajo v števcu in imenovalcu zapisano število, v *Mathematici* ni potrebno uporabiti posebnega ukaza. Ulomek le vnesemo v celico ter pritisnemo SHIFT-ENTER. *Mathematica* vrne okrajšan ulomek.

$$\frac{2257}{3294}$$

$$\frac{37}{54}$$

Naloga ([3], str. 90, nal. 338 č): Izračunajte: $\left(\frac{15}{11}\right)^{-1} : \left(\left(3\frac{1}{6} - 1\frac{7}{9}\right) : \left(\frac{18}{25}\right)^{-1} + 1\frac{18}{23} \cdot 7\frac{2}{3}\right)$.

Izraz zapišemo v »Input« obliki, značilni za program *Mathematica,* in pritisnemo SHIFT-ENTER. *Mathematica* vrne vrednost izraza.

```
1/((15/11)*(((3 + 1/6) - (1 + 7/9))/(18/25)^(-1) +
   (1 + 18/23)*(7 + 2/3)))
```

$$\frac{1}{20}$$

4.3.2. Algebrski ulomki

Pri izrazih z algebrskimi ulomki lahko uporabimo ukaza

- `Together[vsota]`, ki členom v vsoti poišče skupni imenovalec in okrajša ulomek v rezultatu
- `Cancel[ulomek]`, ki okrajša ulomek

Ukaza bomo uporabili v naslednjih nalogah.

Naloga ([3], str. 100, nal. 393e): Izračunajte $\frac{3}{x-1}+\frac{9}{x^2-x-2}+\frac{6}{1-x^2}$.

```
Together[3/(x - 1) + 9/(x^2 - x - 2) + 6/(1 - x^2)]
```

$$\frac{3}{-2+x}$$

Naloga ([3], str. 100, nal. 400): Pokažite, da je vrednost izraza $\frac{\frac{a+3b^2}{b^2}-3}{3-\frac{3a+3b^2}{b^2}}$ neodvisna od vrednosti števil a in b $(a, b \neq 0)$.

```
Cancel[((a + 3*b^2)/b^2 - 3)/(3 - (3*a + 3*b^2)/b^2)]
```

$$-\frac{1}{3}$$

Izraze z algebrskimi ulomki pa lahko seveda poenostavimo tudi s splošnejšim ukazom `Simplify`.

Naloga ([3], str. 94, nal. 360 č): Izračunajte: $\frac{x^{-n+2}-3x^{-n+1}}{x^{2-n}-2x^{1-n}}-\frac{x^{1-n}+x^{-n}}{2x^{-n}-x^{1-n}}$.

```
Simplify[(x^(-n + 2) - 3*x^(-n + 1)) /
  (x^(2 - n) - 2*x^(1 - n)) -
 (x^(1 - n) + x^(-n)) /
  (2/x^n - x^(1 - n))]
```

$$\frac{2\,(-1+x)}{-2+x}$$

```
TraditionalForm[%]
```

$$\frac{2\,(x-1)}{x-2}$$

Rezultat lahko zapišemo v »običajni obliki« (ukaz `TraditionalForm`).

Naloga ([3], str. 100, nal. 394 a, b): Dan je izraz: $\frac{4}{a^2+3a}-\frac{21-a}{9a-a^3}+\frac{a+3}{a^2-3a}$.

a) Poenostavite izraz.

b) Za $a=4$ izračunajte vrednost izraza.

```
Simplify[4/(a^2 + 3*a) - (21 - a)/(9*a - a^3) +
   (a + 3)/(a^2 - 3*a)]
```

$$\frac{6+a}{(-3+a)\,a}$$

```
TraditionalForm[%]
```

$$\frac{a+6}{(a-3)\,a}$$

$\rightarrow$

```
% /. a → 4
```

$$\frac{5}{2}$$

4.4. Realna števila

4.4.1. Intervali

Naloga ([3], str. 112, nal. 425 b): Zapišite presek intervalov: $(-5,-3) \cap (-4,-2)$.

V *Mathematici* zapišemo interval (a, b) kot `Interval[{a,b}]`, nato pa z ukazom `IntervalIntersection` določimo presek danih intervalov.

```
IntervalIntersection[Interval[{-5, -3}], Interval[{-4, -2}]]
```

```
Interval[{-4, -3}]
```

Naloga ([3], str. 112, nal. 426 c): Zapišite unijo intervalov: $[-2, 3] \cup (5, 7)$.

V tem primeru uporabimo ukaz `IntervalUnion`.

```
IntervalUnion[Interval[{-2, 3}], Interval[{5, 7}]]
```

```
Interval[{-2, 3}, {5, 7}]
```

Opomba: Žal v *Mathematici* nismo našli ukaza, ki bi ločil med zaprtim in odprtim intervalom.

4.4.2. Linearne enačbe in neenačbe

Naloga ([3], str. 117, nal. 437): Pokažite, da enačba $\frac{x+1}{(x-1)^2} = \frac{2}{x^2-2x+1} + \frac{x}{x-1}$ nima rešitve.

```
Solve[(x + 1)/(x - 1)^2 == 2/(x^2 - 2 x + 1) + x/(x - 1), x]
```

```
{}
```

Program *Mathematica* vrne izhodni podatek `{}` − prazna množica. To pomeni, da enačba nima rešitve.

Naloga ([3], str. 117, nal. 442 c): Rešite enačbo:

$(3^2 - (2x)^2)^2 + x^2(1 - x)(1 + 2^4x) = 5(3x^3 + 2).$

```
Solve[(3^2 - (2*x)^2)^2 + x^2*(1 - x)*(1 + 2^4*x) == 5*(3*x^3 + 2)]
```

```
{{x → -1}, {x → 1}}
```

Enačba ima dve rešitvi, zapisani sta v obliki seznama.

Naloga ([3], str. 128, nal. 496 a): Rešite sistem enačb:

$$3x + 2y - 4z = 2$$
$$x + 5y + 3z = 18$$
$$5x - 7y - 5z = -2$$

Kot smo že zapisali (poglavje 3.6), lahko z ukazom `Solve` rešujemo tudi sisteme enačb. V prvem zavitem oklepaju z vejicami ločimo enačbe, v drugem pa na enak način naštejemo neznanke. Slednje je obvezno le, če je neznank več kot enačb.

```
Solve[{3 x + 2 y - 4 z == 2, x + 5 y + 3 z == 18, 5 x - 7 y - 5 z == -2}, {x, y, z}]
```

```
{{x → 4, y → 1, z → 3}}
```

Dani sistem enačb ima natanko eno rešitev.

Naloga ([3], str. 132, nal. 509 e): Obravnavajte enačbo glede na vrednosti parametrov a in b iz množice racionalnih števil: $a(ax - 2) = b(bx - 2)$.

Z ukazom `Reduce` lahko obravnavamo enačbe. Najprej navedemo enačbo, nato še neznanko, ki je od enačbe ločena z vejico. Kot smo že zapisali, v programu *Mathematica* znak `||` predstavlja logični ali, znak `&&` pa logični in. Vrednosti parametrov, za katere enačba nima rešitve, program ne izpisuje.

```
Reduce[a (a x - 2) == b (b x - 2), x]
```

```
(a + b ≠ 0 && x == 2/(a + b)) || a == b
```

Iz dobljenega zapisa razberemo, da je $x = \frac{2}{a+b}$ za $a + b \neq 0$, če je $a = b$, pa so rešitve enačbe vsa racionalna števila.

Naloga ([3], str. 132, nal. 513 c): Obravnavajte sistem enačb:

$$x - y + z = 2b$$

$$x + y - z = 2c$$

$$-x + y + z = -2a$$

```
Solve[{x - y + z == 2 b, x + y - z == 2 c, -x + y + z == -2 a}, {x, y, z}]
```

```
{{x → b + c, y → -a + c, z → -a + b}}
```

Za reševanje te naloge nismo uporabili nobenega posebnega ukaza. Ugotovimo, da so *x*, *y* in *z* odvisni od parametrov *a*, *b* in *c*.

Naloga ([3], str. 143, nal. 581 č): Rešite neenačbo: $21 - (3 + 2x)^2 \geq 4(9 - x^2)$.

Tudi neenačbe rešujemo z ukazom `Reduce`.

```
Reduce[21 - (3 + 2*x)^2 ≥ 4*(9 - x^2)]
```

```
x ≤ -2
```

Naloga ([3], str. 143, nal. 583 č): Rešite sistem neenačb:

$$\frac{x+1}{2} - \frac{5x+1}{3} < \frac{1}{6}$$

$$x - \frac{3x+1}{2} - \frac{x-14}{3} \geq 0$$

```
Reduce[{(x + 1)/2 - (5 x + 1)/3 < 1/6, x - (3 x + 1)/2 - (x - 14)/3 ≥ 0}]
```

```
0 < x ≤ 5
```

Naloga ([3], str. 143, nal. 584 a): Poiščite vrednosti *x*, za katere velja:

$$(5(x - 1) > (4 - x)3 + 7) \vee (2x + 3 < 4 - 5(x - 4))$$

Naloga zahteva, da poiščemo unijo dveh intervalov. Najprej z ukazom `Reduce` rešimo obe neenačbi, nato z ukazom `Union` poiščemo unijo obeh intervalov

```
Reduce[5 (x - 1) > (4 - x) 3 + 7]
```

```
x > 3
```

```
Reduce[2 x + 3 < 4 - 5 (x - 4)]
```

```
x < 3
```

```
Union [{x < 3, x > 3}]
```

```
{x > 3, x < 3}
```

Ugotovimo, da sta intervala disjunktna. Danemu pogoju torej ustrezajo vsa realna števila, razen števila 3.

Naloga ([3], str. 143, nal. 586): Obravnavajte neenačbo: $(a^2 + b^2)x + b^2 - a^2 < 2abx$. Najprej za reševanje neenačbe uporabimo ukaz `Reduce`.

$$\text{Reduce}\left[\left(a^2 + b^2\right) x + b^2 - a^2 < 2\, a\, b\, x\right]$$

$$b \in \text{Reals} \;\&\&\; \left(\left(a < b \;\&\&\; x < \frac{a+b}{a-b}\right) \;||\; \left(a > b \;\&\&\; x < \frac{a+b}{a-b}\right)\right)$$

Mathematica v rešitvi ni navedla vrednosti spremenljivke *x*, če je $a = b$. To se zgodi zato, ker pri tem pogoju neenačba nima rešitve. Našo trditev preverimo tako, da namesto parametra *b* na ustrezna mesta v neenačbi vstavimo parameter *a* in s tem zadostimo pogoju $a = b$. Ponovno uporabimo ukaz `Reduce`, ki tokrat vrne logično vrednost `False`:

```
Reduce[(2*a^2)*x + a^2 - a^2 < 2*a^2*x]
```

```
False
```

4.4.3. Absolutna vrednost

Naloga ([3], str. 146, nal. 594 k): Izračunajte: $\left|\left|\sqrt{2} + 4\right| - \sqrt{2}\,\right| + \left|\left|\sqrt{2} - 4\right| - \sqrt{2}\right|$.

Za poenostavitev izraza uporabimo ukaza `Abs[izraz]` za absolutno vrednost in `Sqrt[izraz]` za kvadratni koren.

```
Abs[Abs[Sqrt[2] + 4] - Sqrt[2]] + Abs[Abs[Sqrt[2] - 4] - Sqrt[2]]
```

$8 - 2\sqrt{2}$

Naloga ([3], str. 147, nal. 602 e): Rešite enačbo: $6 + |x - 1| = |3x + 1|$.

```
Solve[6 + Abs[x - 1] == Abs[3 x + 1]]
```

```
{{x → -4}, {x → 2}}
```

Za rešitev naloge smo uporabili že predstavljene ukaze.

4.4.4. Potence z racionalnimi eksponenti in iracionalne enačbe

V tem razdelku bomo zapisali le reševanje dveh nalog, saj večino nalog iz učbenika v tem poglavju rešimo z že znanimi ukazi.

Naloga ([4], str. 153, nal. 528): Poenostavite izraz $(a^{-\frac{3}{2}}b(ab^{-2})^{-\frac{1}{2}}(a^{-1})^{-\frac{2}{3}})^3$ in izračunajte njegovo vrednost za $a = \frac{\sqrt{2}}{2}$ in $b = \frac{1}{\sqrt[3]{2}}$.

Izraz najprej poenostavimo z že znanim ukazom `Simplify`:

```
Simplify[(b/(a^(3/2)*(a/b^2)^(1/2)*(a^(-1))^(2/3)))^3]
```

$$\frac{b^7\sqrt{\frac{a}{b^2}}}{a^{9/2}}$$

Vidimo, da *Mathematica* ne poenostavi izraza popolnoma (izogne se absolutni vrednosti v rezultatu). Če dodamo, da sta *a* in *b* pozitivni števili, je rezultat zapisan v preprostejši obliki.

```
Simplify[(b^7*Sqrt[a/b^2])/a^(9/2), a > 0 && b > 0]
```

$$\frac{b^6}{a^4}$$

Na koncu izrazu priredimo še podani vrednosti za *a* in *b*:

```
% /. {a → √2/2, b → 1/∛2}
```

→

```
1
```

Naloga ([4], str. 156, nal. 546): Pokažite, da je $x_1 = -1$ rešitev enačbe

$$\frac{4}{x+\sqrt{x^2+x}} - \frac{1}{x-\sqrt{x^2+x}} = \frac{3}{x}.$$

Poiščite še drugo rešitev.

Enačbo rešimo z ukazom `Solve`; dobimo dve rešitvi, od katerih je ena zares enaka -1:

```
Solve[4/(x + Sqrt[x^2 + x]) - 1/(x - Sqrt[x^2 + x]) == 3/x]
```

```
{{x → -1}, {x → 9/16}}
```

4.5. Funkcija - osnovni pojmi ter nekatere elementarne funkcije

Ukazi, ki jih uporabljamo za predstavitev osnovnih pojmov v zvezi z različnimi elementarnimi funkcijami, so podobni. Enako velja za risanje grafov različnih funkcij. Zato bomo naloge v zvezi s funkcijami, ki jih obravnavamo v 1. in 2. letniku gimnazijskega programa, predstavili v istem poglavju. Dodajmo še, da smo se omejili le na tiste funkcije, ki smo jih pri rednem pouku spoznali do zaključka raziskovalne naloge.

Naloga ([3], str. 193, nal. 705 a,b): Narišite grafa funkcij f, g: **N**→**N** in zapišite njuni zalogi vrednosti

a) $f(x) = 2x$
b) $g(x) = 3$

Zaloga vrednosti funkcije so v primeru a) soda naravna števila, v primeru b) pa so vse funkcijske vrednosti enake 3. Pri risanju grafa se bomo omejili na prvih 30 točk. Uporabili bomo ukaz `ListPlot`, pri katerem imamo naslednji možnosti:

- ukaz `ListPlot[{y1,y2,y3, …}]` nariše točke, katerih druge koordinate so y_1, y_2, y_3, …, prve koordinate pa so privzeto naravna števila
- ukaz `ListPlot[{{x1,y1},{x2,y2},{x3,y3},…}]` nariše točke (x_1, y_1), (x_2, y_2), (x_3, y_3), …

Seveda lahko pri risanju grafa uporabimo različne možnosti – na primer določimo velikost in barvo točk, njihovo obliko, točke povežemo s koordinatno osjo (ukaz `Filling->Axis`). V našem primeru (in sorodnih primerih) uporabimo tudi ukaz `Table`, in sicer v naslednjih oblikah:

- ukaz `Table[izraz,{n}]` tvori seznam *n* kopij predmeta *izraz*
- ukaz `Table[izraz,{k,n}]` tvori seznam vrednosti *izraza*, pri čemer se *k* spreminja od 1 do *n*
- ukaz `Table[izraz,{k,m,n,d}]` tvori seznam vrednosti *izraza*, pri čemer se *k* spreminja od *m* do *n* v korakih po *d*

Vrnimo se k reševanju naloge.

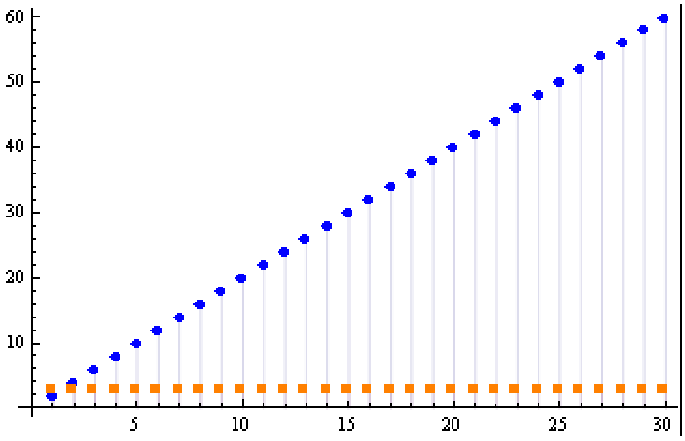
```
sodo = Table[2 n, {n, 30}]
```

```
{2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28,
 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60}
```

```
tri = Table[3, {30}]
```

```
{3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3}
```

```
ListPlot[{sodo, tri}, PlotStyle → {Blue, Orange}, PlotMarkers → Automatic,
 Filling → Axis]
```



Graf 4: Grafa funkcij *f* in *g* s predpisoma $f(x) = 2x$ in $g(x) = 3$ (definicijsko območje smo v obeh primerih omejili na prvih 30 naravnih števil)

Naloga ([3], str. 194, nal. 715): Funkcijo f: **R**→**R**, $f(x) = x^3 - 3x^2 - x + 3$, tabelirajte na intervalu [-4,4] s korakom 1, zapišite ničle ter presečišče grafa z ordinatno osjo.

Najprej definirajmo funkcijo *f*:

```
f[x_] := x^3 - 3*x^2 - x + 3
```

Vrednosti funkcije za izbrane *x*-e dobimo z ukazom:

```
Table[f[x], {x, -4, 4, 1}]
```

```
{-105, -48, -15, 0, 3, 0, -3, 0, 15}
```

Ničle grafa funkcije:

```
Solve[f[x] == 0]
```

```
{{x → -1}, {x → 1}, {x → 3}}
```

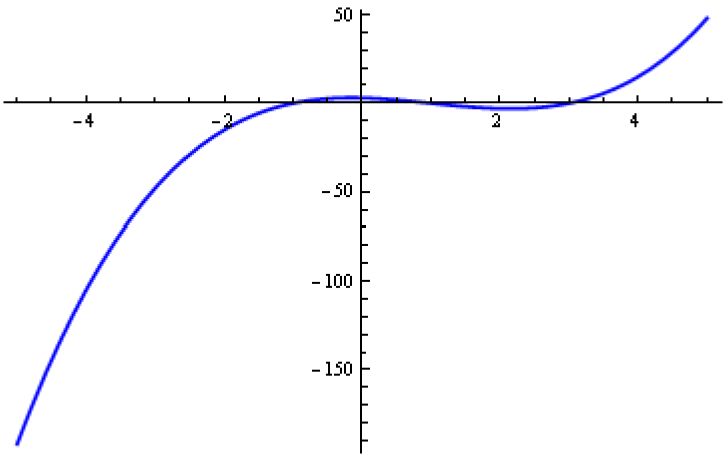
Presečišče grafa funkcije z ordinatno osjo:

```
f[x] /. x → 0
```

```
3
```

Narišimo še graf funkcije *f* na intervalu $[-5,5]$:

```
Plot[f[x], {x, -5, 5}, PlotStyle → {RGBColor[0, 0, 1], Thickness[.005]}]
```



Graf 5: Graf funkcije *f* s predpisom $f(x) = x^3 - 3x^2 - x + 3$

Naloga ([4], str. 188, nal. 625): Dana je funkcija $f(x) = \frac{1}{x^2}$. V isti koordinatni sistem narišite grafa funkcij $g: x \to f(\frac{1}{2}x)$ in $h: x \to f(\frac{1}{2}x - 1)$ ter zapišite enačbi dobljenih krivulj.

V predpisu funkcije *f* nadomestimo *x* z $\frac{1}{2}x$ oziroma z $\frac{1}{2}x - 1$:

```
x^(-2) /. x -> 1/2 x
```

$$\frac{4}{x^2}$$

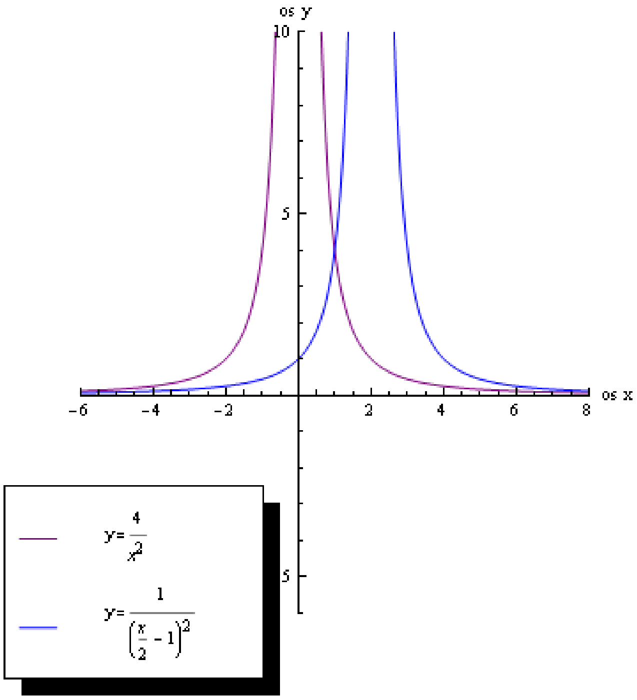
```
x^(-2) /. x -> 1/2 x - 1
```

$$\frac{1}{\left(\frac{x}{2} - 1\right)^2}$$

Grafa dobljenih funkcij narišemo v isti koordinatni sistem. Grafu bomo dodali legendo, zato moramo najprej naložiti paket *Plot Legends*.

```
<< PlotLegends`
```

```
Plot[{4 / x^2, 1 / (x / 2 - 1)^2}, {x, -6, 8}, PlotRange → {{-6, 8}, {-6, 10}},
 AxesLabel → {"os x", "os y"}, AspectRatio → Automatic, PlotStyle → {Purple, Blue},
 PlotLegend → {"y=4/x^2", "y=1/((x/2 - 1)^2)"}]
```

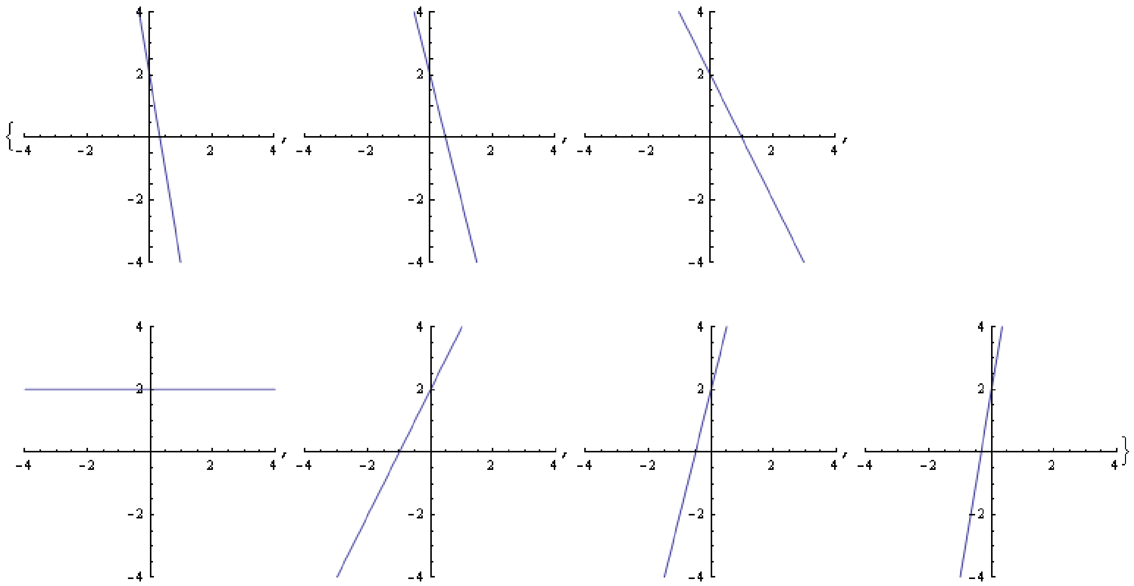


Graf 6: Krivulji z enačbama $y=\frac{4}{x^2}$ **in** $y=\frac{1}{(\frac{x}{2}-1)^2}$

V nadaljevanju si bomo ogledali nekaj primerov v zvezi z linearno funkcijo.

<u>Zgled:</u> Narišimo šop premic z enačbo $y = kx + 2$, pri čemer naj k zavzame vrednosti -6, -4, -2, 0, 2, 4 in 6. Vsako premico bomo najprej narisali v svojem koordinatnem sistemu.

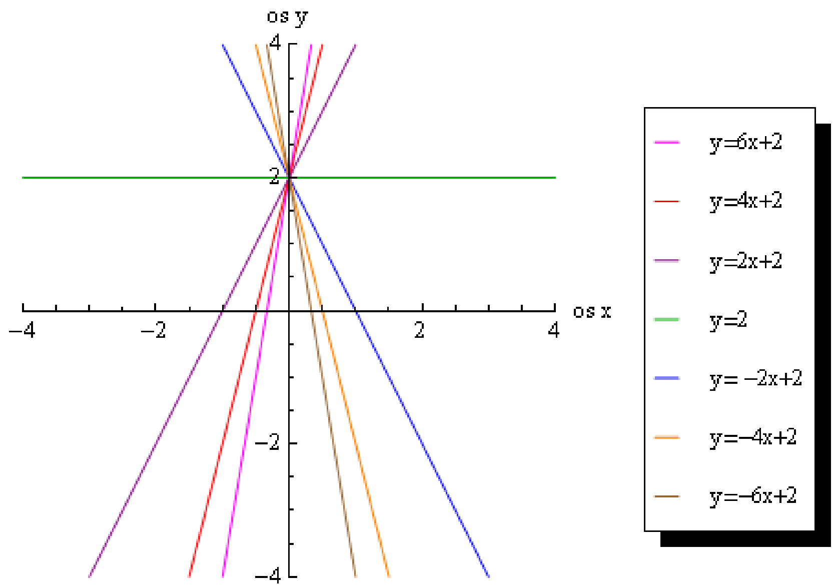
```
Table[Plot[k x + 2, {x, -4, 4}, PlotRange → {{-4, 4}, {-4, 4}}, AspectRatio → Automatic],
 {k, -6, 6, 2}]
```



Graf 7: Premice z enačbo $y = kx + 2$, k = -6, -4, -2, 0, 2, 4, 6

Premice bomo narisali še v istem koordinatnem sistemu.

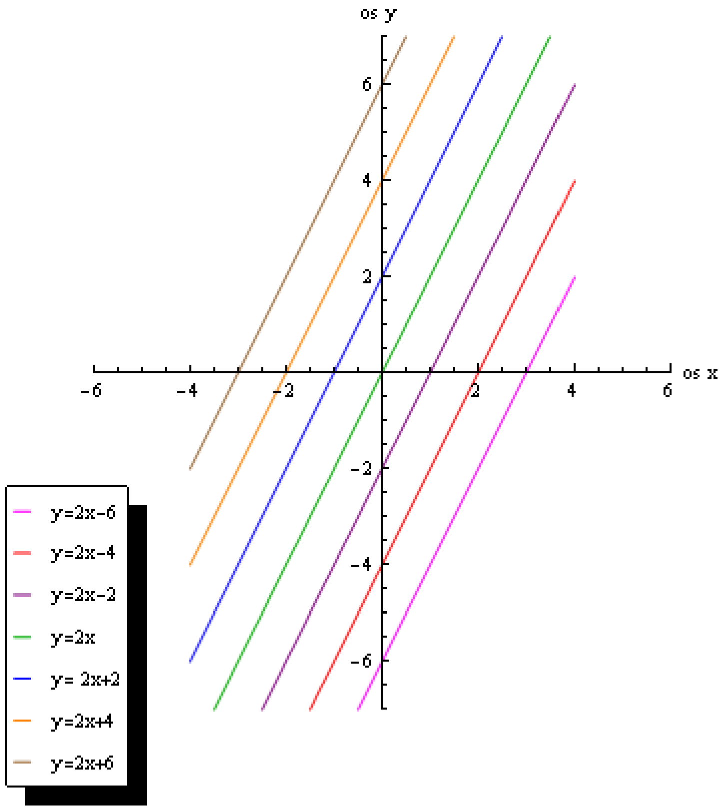
```
Plot[{6 x + 2, 4 x + 2, 2 x + 2, 2, -2 x + 2, -4 x + 2, -6 x + 2}, {x, -4, 4},
 PlotRange → {{-4, 4}, {-4, 4}}, AxesLabel → {"os x", "os y"}, AspectRatio → Automatic,
 PlotStyle → {Magenta, Red, Purple, Darker[Green], Blue, Orange, Brown},
 PlotLegend → {"y=6x+2", "y=4x+2", "y=2x+2", "y=2", "y= -2x+2", "y=-4x+2", "y=-6x+2"},
 LegendPosition → {1.1, -0.8}, LegendSize → 1.4]
```



Graf 8: Šop premic z enačbo $y = kx + 2$, k = -6, -4, -2, 0, 2, 4, 6

Na podoben način narišimo še snop premic z enačbo $y = 2x + n$, pri čemer naj n zavzame vrednosti -6, -4, -2, 0, 2, 4 in 6.

```
Plot[{2 x - 6, 2 x - 4, 2 x - 2, 2 x, 2 x + 2, 2 x + 4, 2 x + 6}, {x, -4, 4},
 PlotRange → {{-6, 6}, {-7, 7}}, AxesLabel → {"os x", "os y"}, AspectRatio → Automatic,
 PlotStyle → {Magenta, Red, Purple, Darker[Green], Blue, Orange, Brown},
 PlotLegend → {"y=2x-6", "y=2x-4", "y=2x-2", "y=2x", "y= 2x+2", "y=2x+4", "y=2x+6"}]
```



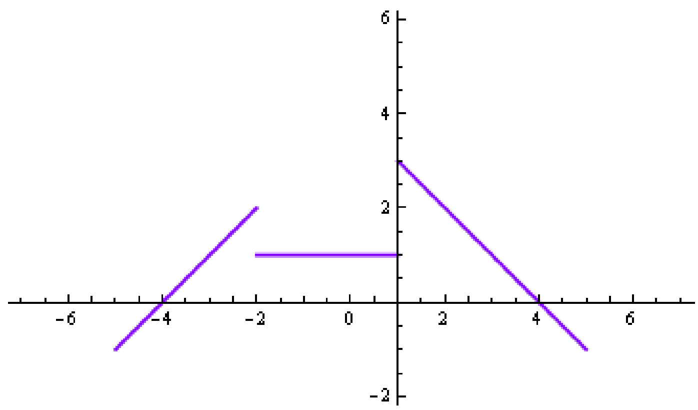
Graf 9: Snop premic z enačbo $y = 2x + n$, n = -6, -4, -2, 0, 2, 4, 6

Naloga ([3], str. 209, nal. 744 a): Narišite graf funkcije $f(x) = \begin{cases} -x+4; \ x > 1 \\ 1; \ -2 < x \leq 1 \\ x+4; \ \ x \leq -2 \end{cases}$

V tej nalogi ima funkcija f različne predpise na različnih intervalih. Graf funkcije f bomo narisali na naslednji način:

```
f1[x_] := -x + 4
f2[x_] := 1
f3[x_] := x + 4
```

```
g1 = Plot[f1[x], {x, 1, 5}, PlotStyle → {RGBColor[0.5, 0, 1], Thickness[.005]}];
g2 = Plot[f2[x], {x, -2, 1}, PlotStyle → {RGBColor[0.5, 0, 1], Thickness[.005]}];
g3 = Plot[f3[x], {x, -5, -2}, PlotStyle → {RGBColor[0.5, 0, 1], Thickness[.005]}];
Show[g1, g2, g3, PlotRange → {{-7, 7}, {-2, 6}}, AspectRatio → Automatic]
```



Graf 10: Graf funkcije *f* s predpisom $f(x) = \begin{cases} -x+4; & x > 1 \\ 1; & -2 < x \leq 1 \\ x+4; & x \leq -2 \end{cases}$

Ob koncu obravnave linearne funkcije dodajmo še naslednji zgled.

Zgled: Označimo množico točk v ravnini, ki je dana s sistemom neenakosti

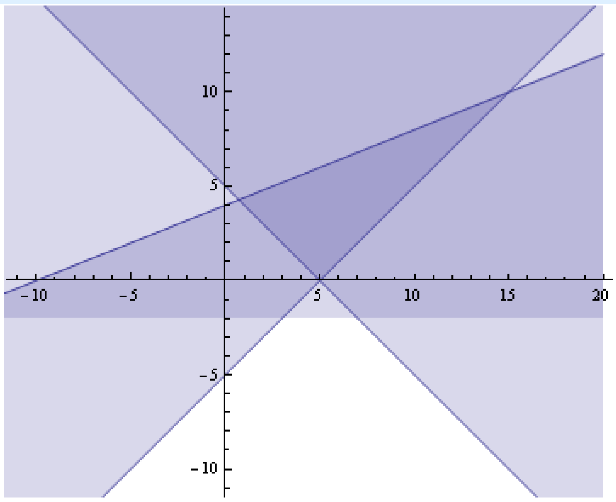
$$x + y \geq 5, \quad x - y \leq 5, \quad y \leq 0{,}4x + 0{,}4.$$

Pri označevanju te množice točk bomo v isti koordinatni sistem narisali ustrezne premice in v vsakem primeru pobarvali izbrano polravnino. Pri tem bomo uporabili ukaza

- `Filling->Top`, ki osenči točke nad krivuljo
- `Filling->Bottom`, ki osenči točke pod krivuljo

Iskana množica točk je presek vseh polravnin.

```
g1 = Plot[5 - x, {x, -15, 20}, Filling → Top];
g2 = Plot[x - 5, {x, -15, 20}, Filling → Top];
g3 = Plot[0.4 x + 4, {x, -15, 20}, Filling → Bottom];
Show[g1, g2, g3, PlotRange → {{-11, 20}, {-11, 14}}, AspectRatio → Automatic]
```



Graf 11: Množica točk v ravnini, dana s sistemom neenakosti $x + y \geq 5, x - y \leq 5, y \leq 0,4x + 0,4$

Oglejmo si še reševanje kompleksnejše naloge iz področja kvadratne funkcije.

Naloga ([4], str. 207, nal. 739): Dana je funkcija $f(x) = -(x-2)^2 + 9$.

a) Poiščite enačbi tangent na parabolo, ki sekata ordinatno os pri 6, ter zapišite dotikališči tangent.
b) V istem koordinatnem sistemu narišite graf funkcije *f* in njeni tangenti.

Ker leži točka (0,6) na tangenti, ima enačba tangente obliko $y = kx + 6$. Poiskali bomo dotikališče tangente na parabolo.

```
Simplify[-(x - 2)^2 + 9 == k x + 6]
```

```
1 + k x + x^2 == 4 x
```

Diskriminanta zgornje kvadratne enačbe mora biti enaka 0 (iščemo namreč eno »dvojno« rešitev enačbe):

```
Solve[(k - 4)^2 - 4 == 0, k]
```

```
{{k → 2}, {k → 6}}
```

Ustrezni sta dve tangenti z enačbama $y = 2x + 6$ in $y = 6x + 6$. Določimo dotikališči tangent na paraboli:

```
Reduce[-(x - 2)^2 + 9 == 2 x + 6]
```

```
x == 1
```

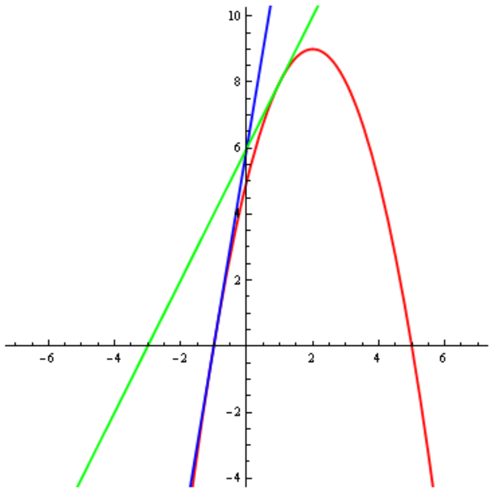
```
Reduce[-(x - 2)^2 + 9 == 6 x + 6]
```

```
x == -1
```

Dotikališči tangent sta torej točki $T_1(1,8)$ in $T_2(-1,0)$.
Narišimo še parabolo in njeni tangenti:

```
f1[x_] := -(x - 2)^2 + 9
f2[x_] := 6 x + 6
f3[x_] := 2 x + 6
```

```
g1 = Plot[f1[x], {x, -6, 6}, PlotStyle → {RGBColor[1, 0, 0], Thickness[.005]}];
g2 = Plot[f2[x], {x, -6, 6}, PlotStyle → {RGBColor[0, 0, 1], Thickness[.005]}];
g3 = Plot[f3[x], {x, -6, 6}, PlotStyle → {RGBColor[0, 1, 0], Thickness[.005]}];
Show[g1, g2, g3, PlotRange → {{-7, 7}, {-4, 10}}, AspectRatio → Automatic]
```



Graf 12: Graf funkcije f s predpisom $f(x) = -(x-2)^2 + 9$ in njeni tangenti, ki sekata ordinatno os pri 6

4.6. Osnove statistike

Naloga ([3], str. 164, nal. 636): Avto-moto društvo je merilo hitrost odziva posameznih servisnih enot pri izvajanju hitre pomoči. Izberite ustrezno število frekvenčnih razredov in grupirajte rezultate, ki so jih namerili v 30 servisnih enotah, v tabelo frekvenc in relativnih frekvenc.

87,5	62,2	71,1	84,3	69,1	68,0	83,3
51,4	63,5	66,2	50,2	81,5	73,4	62,1
65,3	49,4	52,1	81,2	66,5	60,4	47,5
67,3	72,7	58,1	55,3	59,2	63,7	62,8
70,5	48,7					

Oblikujte tudi tabelo kumulativnih frekvenc.

<u>Opomba:</u> Ker v nalogi ni navedena merska enota rezultatov, bomo privzeli, da je Avto-moto društvo servisnim enotam podelilo točke za hitrost odziva glede na določene kriterije.

Na primeru zapisane naloge bomo s programom *Mathematica* predstavili različne ukaze, s katerimi bomo rešili nalogo in prikazali še druge možnosti statistične obdelave podatkov. Rezultate bomo najprej zapisali v seznam.

```
podatki = {87.5, 62.2, 71.1, 84.3, 69.1, 68.0, 83.3, 51.4, 63.5, 66.2, 50.2, 81.5,
  73.4, 62.1, 65.3, 49.4, 52.1, 81.2, 66.5, 60.4, 47.5, 67.3, 72.7, 58.1, 55.3,
  59.2, 63.7, 62.8, 70.5, 48.7}
```

```
{87.5, 62.2, 71.1, 84.3, 69.1, 68., 83.3, 51.4, 63.5, 66.2, 50.2, 81.5, 73.4, 62.1, 65.3,
 49.4, 52.1, 81.2, 66.5, 60.4, 47.5, 67.3, 72.7, 58.1, 55.3, 59.2, 63.7, 62.8, 70.5, 48.7}
```

Če želimo preveriti število rezultatov ter določiti najmanjšo in največjo vrednost med njimi, uporabimo naslednje funkcije (ukaze):

```
Length[podatki]
```

```
30
```

```
Min[podatki]
```

```
47.5
```

```
Max[podatki]
```

```
87.5
```

V nadaljevanju si oglejmo ukaze, ki jih uporabljamo pri razvrstitvi podatkov v razrede:

- `BinCounts[{x1,x2,x3, …},{xmin,xmax,1}]` prešteje, koliko podatkov x_i je v zaporednih celoštevilskih razredih od najnižjega razreda x_{min} do najvišjega razreda x_{max}
- `BinCounts[{x1,x2,x3, …},dx]` prešteje, koliko podatkov x_i je v zaporednih razredih širine dx
- `BinCounts[{x1,x2,x3, …},[{b1,b2,b3, …}]` prešteje, koliko podatkov x_i je v zaporednih razredih, ki jih predstavimo z intervali $[b_1,b_2)$, $[b_2,b_3)$, $[b_3,b_4)$, …
- `BinCounts[{x1,x2,x3,…},{{-Infinity,c1,c2,c3,…,ck, Infinity}}]` prešteje, koliko podatkov x_i je v zaporednih razredih, ki jih predstavimo z intervali $(-\infty, c_1)$, $[c_1,c_2)$, $[c_2,c_3)$, …, $[c_k,\infty)$

Navedene ukaze bomo uporabili v naši nalogi.

```
BinCounts[podatki, {47, 88, 1}]
```

```
{1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 3, 2, 0,
 1, 2, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 2, 0, 1, 1, 0, 0, 1}
```

Ugotovimo, da v nekaterih celoštevilskih razredih ni nobenega rezultata in da so v posameznem razredu največ 3 rezultati.

Vpeljimo naslednje pojme in njihove oznake:

- f_k – frekvenca *k*-tega razreda (število statističnih enot *k*-tega razreda)
- f_k^* – relativna frekvenca *k*-tega razreda (delež statističnih enot *k*-tega razreda)
- F_k – kumulativna frekvenca *k*-tega razreda (število statističnih enot do vključno s *k*-tim razredom)
- F_k^* – relativna kumulativna frekvenca *k*-tega razreda (delež statističnih enot do vključno s *k*-tim razredom)

Poglejmo, kako zgornje količine izračunamo v naši nalogi. Pred tem bomo rezultate razporedili v razrede širine 5 in izračunali frekvence posameznih razredov:

```
BinCounts[podatki, 5]
```

```
{3, 3, 3, 6, 6, 4, 0, 4, 1}
```

Uporabimo pa lahko tudi drugo možnost. Z naslednjim ukazom bomo rezultate razporedili v polzaprte intervale [47,51), [51,55), …, [83,88) in določili frekvence tako nastalih razredov.

```
BinCounts[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}}]
(*frekvence razredov*)
```

```
{4, 2, 2, 5, 5, 4, 3, 0, 2, 3}
```

Ugotovimo, da se število razredov in frekvence razlikujejo od prejšnje razporeditve rezultatov v razrede. Glede na zadnjo razporeditev izračunajmo relativne frekvence razredov:

```
frekvence = {4, 2, 2, 5, 5, 4, 3, 0, 2, 3}
```

```
{4, 2, 2, 5, 5, 4, 3, 0, 2, 3}
```

```
N[frekvence/30, 2](*relativne frekvence razredov*)
```

```
{0.13, 0.067, 0.067, 0.17, 0.17, 0.13, 0.10, 0, 0.067, 0.10}
```

Za določanje kumulativnih frekvenc razredov bomo uporabili ukaz `Accumulate[list]`, ki vrne seznam enake dolžine kot `list`, v njem pa so po vrsti zapisane delne vsote `lista`.

```
Accumulate[frekvence] (*kumulativne frekvence razredov*)
```

```
{4, 6, 8, 13, 18, 22, 25, 25, 27, 30}
```

Izračunajmo še relativne kumulativne frekvence:

```
kumulative = {4, 6, 8, 13, 18, 22, 25, 25, 27, 30}
```

```
{4, 6, 8, 13, 18, 22, 25, 25, 27, 30}
```

```
N[kumulative/30, 2] (*relativne kumulativne frekvence razredov*)
```

```
{0.13, 0.20, 0.27, 0.43, 0.60, 0.73, 0.83, 0.83, 0.90, 1.0}
```

Zanima nas še, koliko rezultatov je manjših od 50, med 50 in 80 in večjih od 80:

```
BinCounts[podatki, {{-Infinity, 50, 80, Infinity}}]
```

```
{3, 22, 5}
```

Izračune, ki jih zahteva naša naloga, zberimo v tabeli:

Tabela 2: Razporeditev rezultatov (podatkov) naloge 636 iz [3] v razrede in pripadajoče frekvence, relativne frekvence, kumulativne frekvence in relativne kumulativne frekvence

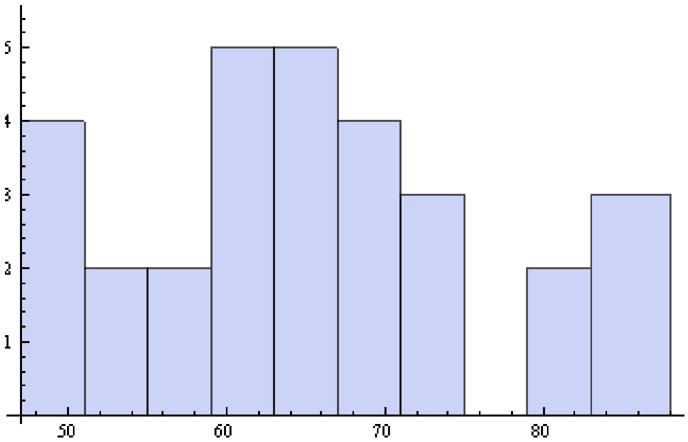
k-ti razred	meje razreda	f_k	f_k^*	F_k	F_k^*
1	47 do 51	4	0,13	4	0,13
2	51 do 55	2	0,07	6	0,20
3	55 do 59	2	0,07	8	0,27
4	59 do 63	5	0,17	13	0,43
5	63 do 67	5	0,17	18	0,60
6	67 do 71	4	0,13	22	0,73
7	71 do 75	3	0,10	25	0,83
8	75 do 79	0	0,00	25	0,83
9	79 do 83	2	0,07	27	0,90
10	83 do 88	3	0,10	30	1,00

Zbrane rezultate želimo predstaviti grafično. Oglejmo si nekatere možnosti, ki jih ponuja *Mathematica*. Ukaz

```
Histogram[{x1,x2,x3, …},{b1,b2,b3, …}]
```

nariše histogram privzete oblike (določena barva pravokotnikov, brez legende, brez zapisov ob oseh, brez poimenovanja grafičnega prikaza) glede na izbrane podatke (frekvence in meje razredov). Na vodoravni osi so prikazane meje razredov, na navpični osi pa frekvence razredov.

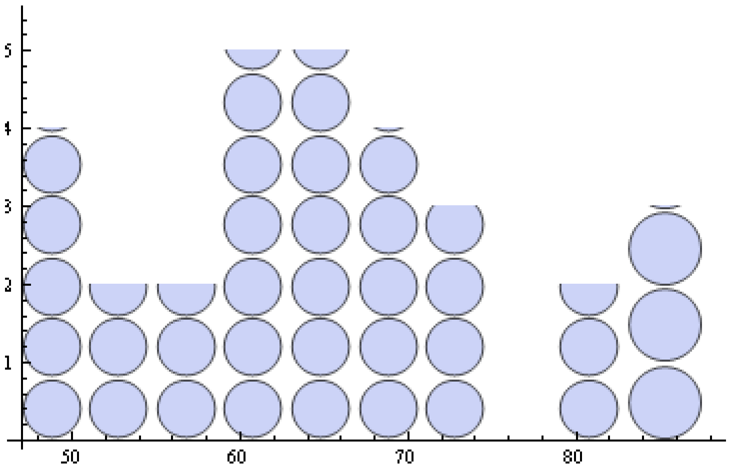
```
Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}}]
```



Graf 13: Histogram za rezultate iz naloge 636 iz [3] – prikaz s pravokotniki

Namesto navpičnih pravokotnikov lahko uporabimo tudi druge oblike prikaza, na primer:

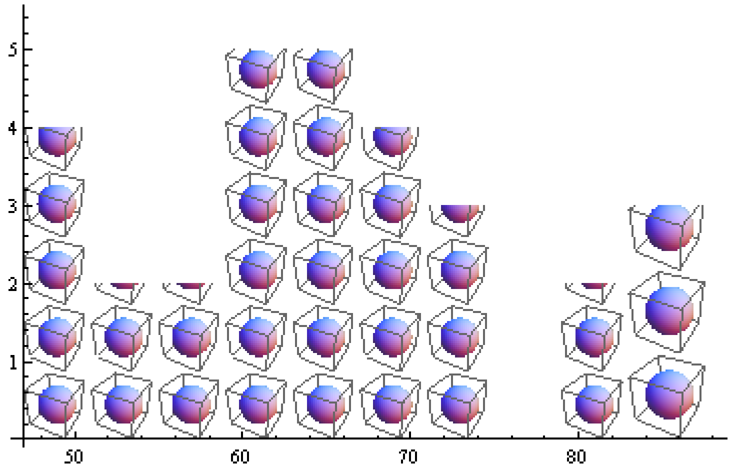
```
Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}},
 ChartElements → Graphics[Disk[]]]
```



Graf 14: Histogram za rezultate iz naloge 636 iz [3] – prikaz s krogi

ali

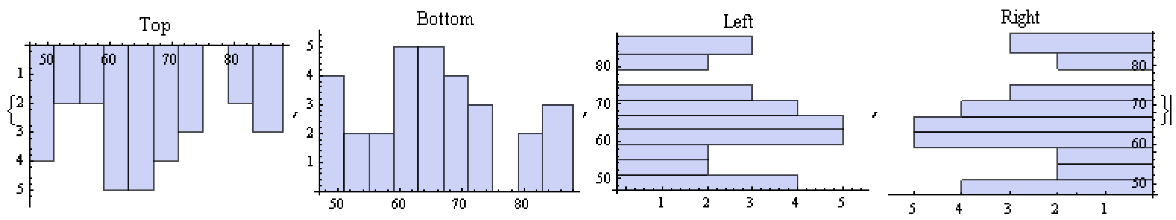
```
Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}},
 ChartElements → Graphics3D[Sphere[]]]
```



Graf 15: Histogram za rezultate iz naloge 636 iz [3] – prikaz s sferami

S spodnjim ukazom pridemo do histogramov, v katerih »rastejo« stolpci od zgoraj navzdol, od spodaj navzgor, z leve ali z desne strani:

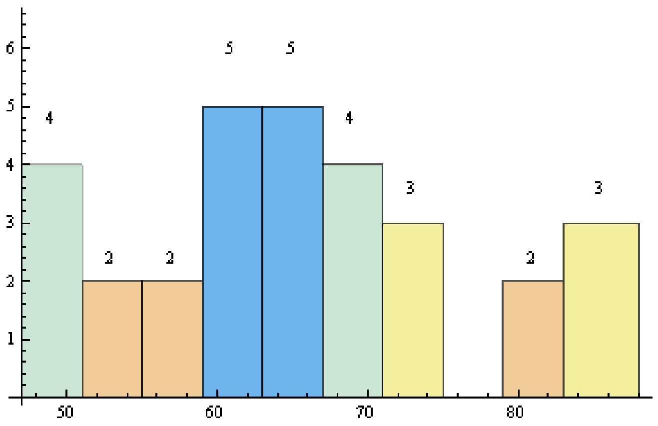
```
Table[Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}}, BarOrigin → o,
  PlotLabel → o], {o, {Top, Bottom, Left, Right}}]
```



Graf 16: Različni histogrami za rezultate iz naloge 636 iz [3] – stolpci »rastejo« od zgoraj navzdol, od spodaj navzgor, z leve proti desni in z desne proti levi

V histogramu lahko izberemo tudi barve pravokotnikov in zapišemo frekvence razredov nad stolpci:

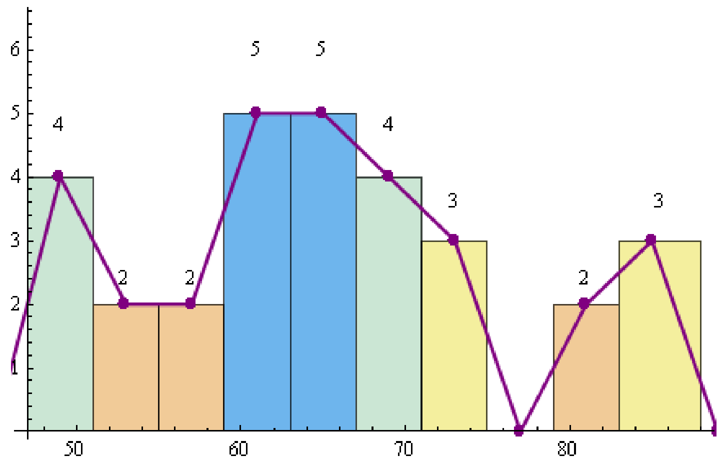
```
Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}},
 LabelingFunction → {0.5, 1.2}, ChartStyle → Red, ColorFunction → "Pastel"]
```



Graf 17: Histogram za rezultate iz naloge 636 iz [3] s pripadajočimi razrednimi frekvencami

Narišimo še frekvenčni poligon. To je lomljena črta, sestavljena iz daljic, katerih krajišča so točke z dvema koordinatama. Prva koordinata pomeni sredino posameznega razreda, druga pa frekvenco tega razreda.

```
g1 = Histogram[podatki, {{47, 51, 55, 59, 63, 67, 71, 75, 79, 83, 88}},
   LabelingFunction → {0.5, 1.2}, ChartStyle → Red, ColorFunction → "Pastel"];
g2 = ListLinePlot[{{45, 0}, {49, 4}, {53, 2}, {57, 2}, {61, 5}, {65, 5},
    {69, 4}, {73, 3}, {77, 0}, {81, 2}, {85, 3}, {89, 0}},
   PlotMarkers → Automatic, PlotStyle → {Purple, Thickness[0.005]}];
Show[g1, g2]
```

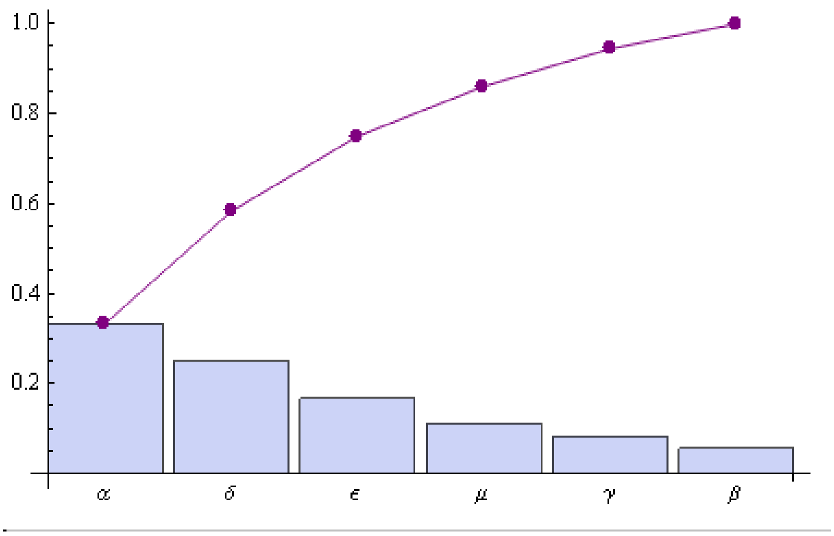


Graf 18: Histogram in frekvenčni poligon za rezultate iz naloge 636 iz [3] s pripadajočimi razrednimi frekvencami

Na tem mestu bomo predstavili še en zanimiv »statistični« ukaz `ParetoPlot`. Ta ukaz lahko uporabljamo le, če smo predhodno naložili paket *Statistical Plots Package*. *Mathematica* z omenjenim ukazom za dani seznam podatkov nariše histogram (pri tem so razredi urejeni od tistega z največjo do tistega z najmanjšo frekvenco), pod vsak stolpec zapiše, kateri podatek predstavlja, hkrati pa nariše lomljeno črto, iz katere so razvidni deleži posameznega podatka v celotnem seznamu in relativne kumulativne frekvence (ogiva). Ukaz `ParetoPlot` bomo uporabili najprej za izmišljen seznam podatkov:

grško = $\{\alpha, \alpha, \epsilon, \alpha, \gamma, \mu, \delta, \alpha, \epsilon, \beta, \delta, \alpha, \alpha, \alpha, \beta, \mu, \delta, \alpha, \gamma, \alpha, \delta, \epsilon, \alpha, \alpha, \delta, \gamma, \delta, \epsilon, \delta, \delta, \epsilon, \epsilon, \mu, \alpha, \mu, \delta\}$

```
ParetoPlot[{α, α, ϵ, α, γ, μ, δ, α, ϵ, β, δ, α, α, α, β, μ, δ, α, γ, α, δ,
  ϵ, α, α, δ, γ, δ, ϵ, δ, δ, ϵ, ϵ, μ, α, μ, δ}, PlotStyle -> Purple]
```



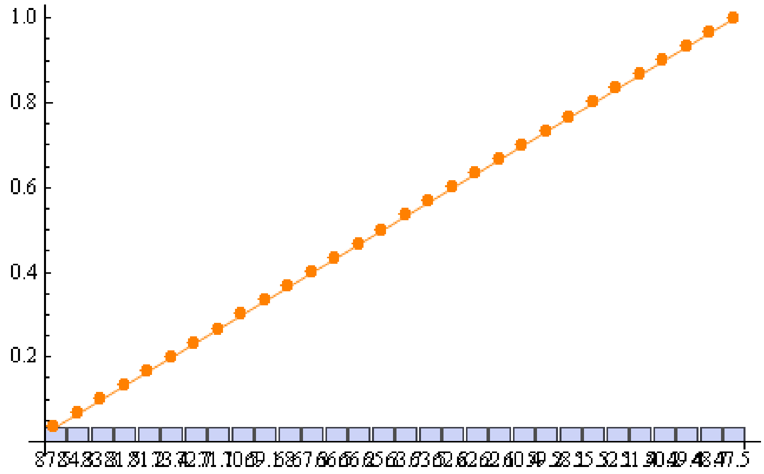
Graf 19: Histogram in ogiva za podatke »grško«

Iz grafičnega prikaza na primer razberemo, da je med podatki po številu na drugem mestu simbol δ, najmanj je simbolov β in da je delež simbolov α med 0,30 in 0,35.

Opomba: Ukaz `ParetoPlot` je v vnosni vrstici zapisan rdeče, ker se uporablja v dveh kontekstih (`StatisticalPlots` in `Global`).

Ukaz `ParetoPlot` uporabimo še za rezultate iz naše naloge. Ker se je vsak rezultat pojavil le enkrat, grafični prikaz ni tako zanimiv.

```
ParetoPlot[podatki, PlotStyle -> Orange]
```



Graf 20: Histogram in ogiva za rezultate iz naloge 636 iz [3]

Nadaljujemo z našo nalogo, ki jo bomo nekoliko dopolnili. Zamislimo si, da bomo po pet zaporednih podanih rezultatov naše naloge po vrsti pripisali mestom Murska Sobota, Maribor, Celje, Ljubljana, Kranj in Koper.

```
podatki1 = {87.5, 62.2, 71.1, 84.3, 69.1}(*Murska Sobota*)
```

```
{87.5, 62.2, 71.1, 84.3, 69.1}
```

```
podatki2 = {68.0, 83.3, 51.4, 63.5, 66.2}(*Maribor*)
```

```
{68., 83.3, 51.4, 63.5, 66.2}
```

```
podatki3 = {50.2, 81.5, 73.4, 62.1, 65.3}(*Celje*)
```

```
{50.2, 81.5, 73.4, 62.1, 65.3}
```

```
podatki4 = {49.4, 52.1, 81.2, 66.5, 60.4}(*Ljubljana*)
```

```
{49.4, 52.1, 81.2, 66.5, 60.4}
```

```
podatki5 = {47.5, 67.3, 72.7, 58.1, 55.3}(*Kranj*)
```

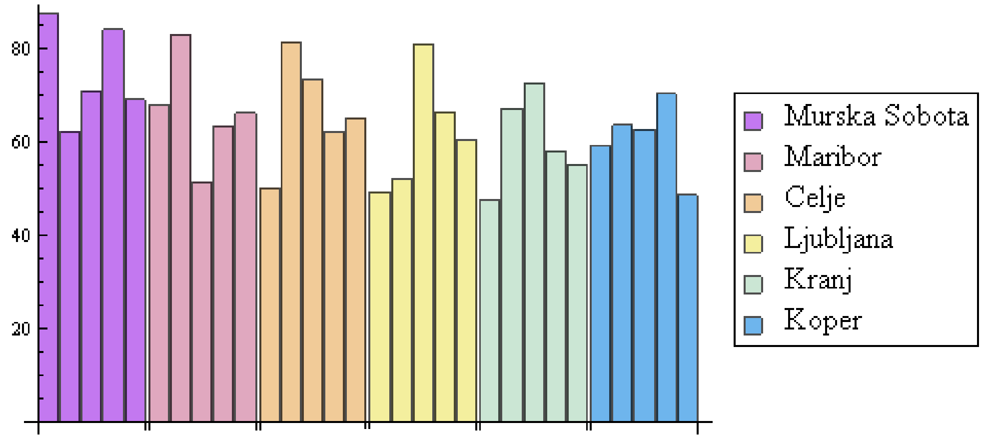
```
{47.5, 67.3, 72.7, 58.1, 55.3}
```

```
podatki6 = {59.2, 63.7, 62.8, 70.5, 48.7}(*Koper*)
```

```
{59.2, 63.7, 62.8, 70.5, 48.7}
```

Naslednji histogram prikazuje rezultate po izbranih mestih:

```
BarChart[{podatki1, podatki2, podatki3, podatki4, podatki5, podatki6},
 ChartLegends → {{"Murska Sobota", "Maribor", "Celje", "Ljubljana", "Kranj", "Koper"},
   None}, ChartStyle → {"Pastel", None}]
```

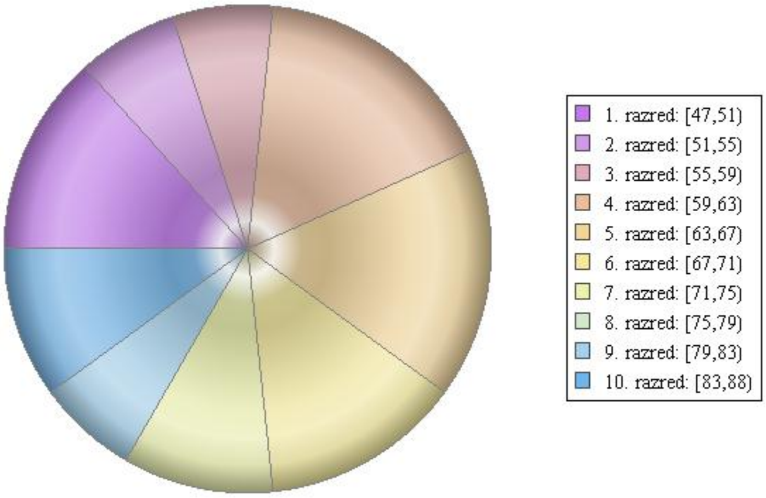


Graf 21: Histogram za rezultate iz dopolnjene naloge 636 iz [3] za izmišljenih šest mest

Iz prikazanega histograma lahko na primer razberemo, v katerem mestu je bil dosežen najboljši oziroma najslabši rezultat, prav tako lahko na enostaven način izvedemo primerjavo uspeha po mestih.

Vrnimo se spet k danim podatkom naše naloge. Deleže rezultatov po posameznih razredih bomo predstavili še s tortnim diagramom oziroma strukturnim krogom:

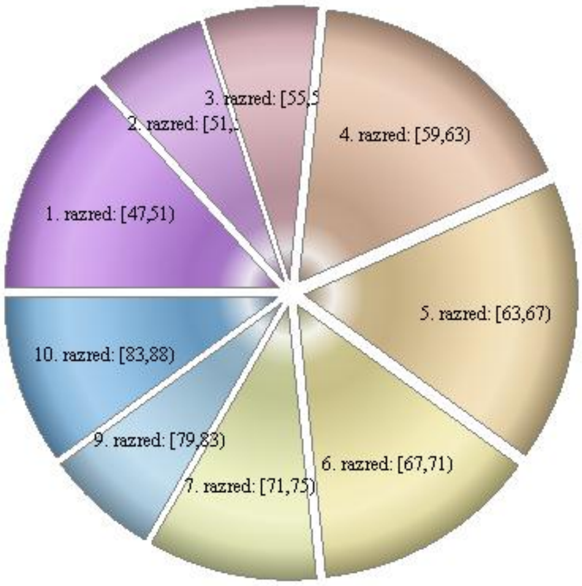
```
PieChart[{4, 2, 2, 5, 5, 4, 3, 0, 2, 3}, ChartElementFunction → "GlassSector",
 ChartStyle → "Pastel",
 ChartLegends → {"1. razred: [47,51)", "2. razred: [51,55)",
   "3. razred: [55,59)", "4. razred: [59,63)", "5. razred: [63,67)",
   "6. razred: [67,71)", "7. razred: [71,75)", "8. razred: [75,79)",
   "9. razred: [79,83)", "10. razred: [83,88)"}]
```



Graf 22: Strukturni krog za rezultate iz naloge 636 iz [3] – krožni izseki se stikajo

Ali na primer:

```
PieChart[{4, 2, 2, 5, 5, 4, 3, 0, 2, 3}, ChartElementFunction → "GlassSector",
 ChartStyle → "Pastel", SectorSpacing → {Tiny, None},
 ChartLabels → {"1. razred: [47,51)", "2. razred: [51,55)",
   "3. razred: [55,59)", "4. razred: [59,63)", "5. razred: [63,67)",
   "6. razred: [67,71)", "7. razred: [71,75)", "8. razred: [75,79)",
   "9. razred: [79,83)", "10. razred: [83,88)"}]
```



Graf 23: Strukturni krog za rezultate iz naloge 636 iz [3] – krožni izseki so razmaknjeni

Za rezultate iz naše naloge bomo izračunali tudi srednje vrednosti – to so značilne vrednosti, s katerimi lahko opišemo celotno populacijo. Izračunali bomo:

- *aritmetično sredino* (povprečje); ukaz: `Mean[seznam]`
- *mediano* (vrednost, ki leži točno na sredi rangirne vrste - vrste, v kateri so podatki razvrščeni od najmanjše do največje vrednosti); ukaz: `Median[seznam]`
- *modus* (najpogostejša vrednost med podatki); ukaz: `Commonest[seznam]`
- *kvartile* (vrednosti, ki delijo rangirno vrsto na štiri enake dele; tako ima 25 % podatkov manjšo in 75 % podatkov večjo vrednost od prvega kvartila, drugi kvartil sovpada z mediano, za tretji kvartil pa velja, da ima od njega 75 % podatkov manjšo, 25 % podatkov pa večjo vrednost); ukaz: `Quartiles[seznam]`

```
Mean[podatki] (*aritmetična sredina*)
```

```
65.15
```

```
Median[podatki] (*mediana*)
```

```
64.5
```

```
Commonest[podatki] (*modus*)
```

```
{87.5, 62.2, 71.1, 84.3, 69.1, 68., 83.3, 51.4, 63.5,
 66.2, 50.2, 81.5, 73.4, 62.1, 65.3, 49.4, 52.1, 81.2, 66.5,
 60.4, 47.5, 67.3, 72.7, 58.1, 55.3, 59.2, 63.7, 62.8, 70.5, 48.7}
```

Določevanje modusa v našem primeru ni zanimivo, saj vsak rezultat nastopi le enkrat.

Oglejmo si določevanje modusa za prej zapisan izmišljen seznam.

```
Commonest[{α, α, ϵ, α, γ, μ, δ, α, ϵ, β, δ, α, α, α, β, μ, δ, α, γ, α, δ,
  ϵ, α, α, δ, γ, δ, ϵ, δ, δ, ϵ, ϵ, μ, α, μ, δ}]
```

```
{α}
```

Z ukazom `Commonest[seznam,n]` pridemo do *n* najpogostejših vrednosti v seznamu.

```
Commonest[{α, α, ϵ, α, γ, μ, δ, α, ϵ, β, δ, α, α, α, β, μ, δ, α, γ, α, δ,
  ϵ, α, α, δ, γ, δ, ϵ, δ, δ, ϵ, ϵ, μ, α, μ, δ}, 2]
```

```
{α, δ}
```

Rezultat se ujema s prikazom grafa 19.

Vrnimo se spet k naši nalogi in izračunajmo kvartile:

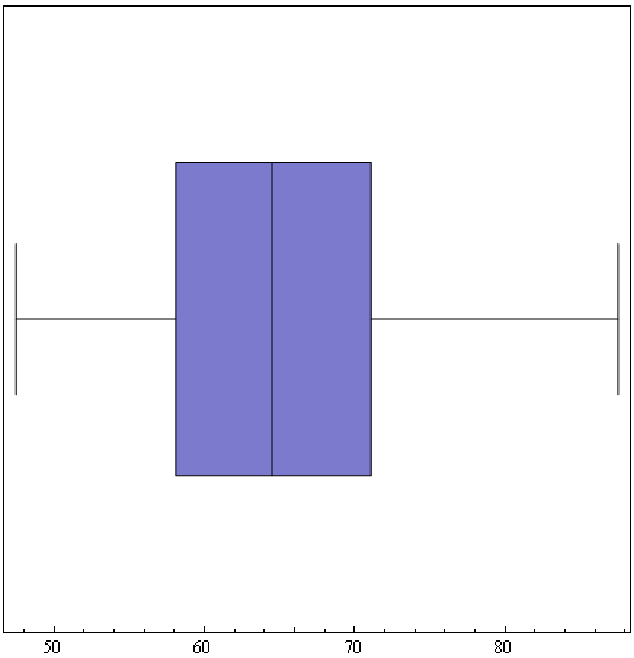
```
Quartiles[podatki] (*prvi, drugi in tretji kvartil*)
```

```
{58.1, 64.5, 71.1}
```

Opazimo, da drugi kvartil in mediana zares sovpadata.

Z ukazom `BoxWhiskerPlot[seznam]` pa narišemo še diagram kvartilov, iz katerega razberemo minimalno in maksimalno vrednost podatkov ter vse tri kvartile.

```
BoxWhiskerPlot[podatki, BoxOrientation → Horizontal]
(*diagram kvartilov*)
```



Graf 24: Diagram kvartilov za rezultate iz naloge 636 iz [3]

Včasih nas razen srednjih vrednosti zanima tudi razpršenost podatkov. Določamo jo na več načinov, najpogostejši pa so:

- *variacijski razmik* (razlika med maksimalno in minimalno vrednostjo podatkov); ukaz: `Max[seznam]- Min[seznam]`
- *interkvartilni rang* (razlika med tretjim in prvim kvartilom; ukaz: `InterquartileRange[seznam]`
- *varianca* (povprečna vrednost kvadratov odmikov od aritmetične sredine); ukaz: `Variance[seznam]`
- *standardna deviacija* ali *standardni odklon* (kvadratni koren iz variance); ukaz: `StandardDeviation[seznam]`

Za rezultate iz naše naloge dobimo naslednje mere razpršenosti:

```
Max[podatki] - Min[podatki] (*variacijski razmik*)
```

```
40.
```

```
InterquartileRange[podatki] (*interkvartilni rang*)
```

```
13.
```

```
Variance[podatki] (*varianca*)
```

```
123.692
```

```
StandardDeviation[podatki] (*standardna deviacija*)
```

```
11.1217
```

Pri raziskovanju uporabljenih ukazov, predstavljenih v 3. in 4. poglavju, smo si pomagali z viri [1], [2], [7] in [8].

5. METODOLOGIJA

Z namenom, da odgovorimo na zastavljena raziskovalna vprašanja ter potrdimo ali ovržemo zastavljene hipoteze, smo uporabili dve raziskovalni metodi:

- proučevanje in uporaba ukazov v programu *Mathematica*
- anketa

Uporaba prve metode je podrobno predstavljena v 3. in 4. poglavju raziskovalne naloge, ki predstavljata njen teoretični del. V poglavju »Razprava« bomo na osnovi te metode odgovorili na prvo in drugo raziskovalno vprašanje ter potrdili ali ovrgli prvi dve zastavljeni hipotezi. V naslednjem poglavju bomo predstavili drugo raziskovalno metodo – anketo.

6. ANKETA

Dijakom enega oddelka 2. letnika naše šole smo predstavili uporabo programa *Mathematica* v uri ponavljanja in utrjevanja iz področja opisne statistike. Anketni vprašalnik se je delno nanašal na to predstavitev, zato so ga izpolnili le dijaki omenjenega razreda. Izpolnjevanje je potekalo 5. 11. 2012 v okviru pouka matematike.

Anketni vprašalnik je vseboval 12 vprašanj. Večina odgovorov je bila ponujena, pri enem vprašanju so lahko dijaki dodali tudi svoj odgovor. Pri treh vprašanjih so dijaki podali svojo oceno iz lestvice od 1 (nezanimivo) do 5 (zelo zanimivo).

6.1. Vprašalnik

Anketni vprašalnik vsebuje 12 vprašanj. Zaradi večje preglednosti vsebujejo vprašanja le glagole za moški spol, nanašajo pa se na dijakinje in dijake. Nekatera vprašanja so splošna, druga pa so v povezavi s predstavitvijo učne ure z uporabo programa *Mathematica,* ki si jo pravkar videl. Prosimo te, da na vprašanja odgovoriš iskreno.

1. Spol: moški ženski

2. Kakšna je bila tvoja zaključena ocena pri matematiki v 1. letniku?
1 2 3 4 5

3. Ali razmišljaš o študiju, povezanem z matematiko?
Da Ne

4. Si pred današnjo uro matematike že kdaj slišal za program *Mathematica*?
Da Ne

5. Če si na prejšnje vprašanje odgovoril z Da, ali si program že kdaj uporabljal?
Da Ne

6. Bi uporabo programa *Mathematica* vključil v redni pouk, če bi imel to možnost?
Da Ne

7. Misliš, da bi bil za delo pri pouku matematike bolj motiviran, če bi le-ta vključeval uporabo programa *Mathematica*?

Da Ne

8. Na lestvici od 1 (nezanimiva) do 5 (zelo zanimiva) označi, kako zanimiva se ti je zdela uporaba programa.

1 2 3 4 5

9. Pri katerem od naštetih področij se ti zdi, da bi program *Mathematica* bil najbolj uporaben? Izbereš lahko en ali dva odgovora.

a) Poenostavljanje izrazov
b) Reševanje enačb in neenačb
c) Risanje grafov funkcij
d) Obdelava podatkov (statistika)
e) Drugo ______________________

10. Si mnenja, da bi ob predpostavki osnovnega predznanja uporabe programa nalogo rešil hitreje s programom *Mathematica*, kot če bi jo reševal 'peš'?

Da Ne

11. Na lestvici od 1 (sploh ne) do 5 (zelo) označi, ali bi želel uporabljati program *Mathematica* doma?

1 2 3 4 5

12. Na lestvici od 1 (sploh ne) do 5 (zelo) označi, ali si se pripravljen naučiti uporabljati program *Mathematica*?

1 2 3 4 5

Hvala za sodelovanje. Rezultate ankete si boš lahko ogledal v okviru raziskovalne naloge »*Mathematica* pri matematiki v 1. in 2. letniku splošnega gimnazijskega programa«.

6.2. Prikaz in analiza rezultatov ankete

Vprašalnik je izpolnilo 28 anketirancev, od tega 12 dijakinj (43 %) in 16 dijakov (57 %). Odgovore na vprašanja smo prikazali na različne načine:

- s tabelami, v katerih so predstavljena števila posameznih odgovorov,
- z grafičnimi prikazi, kjer smo uporabili tortni in stolpični diagram.

Pri nekaterih vprašanjih smo naredili tudi primerjavo med odgovori dijakov glede na spol in glede na zaključno oceno pri matematiki v 1. letniku.

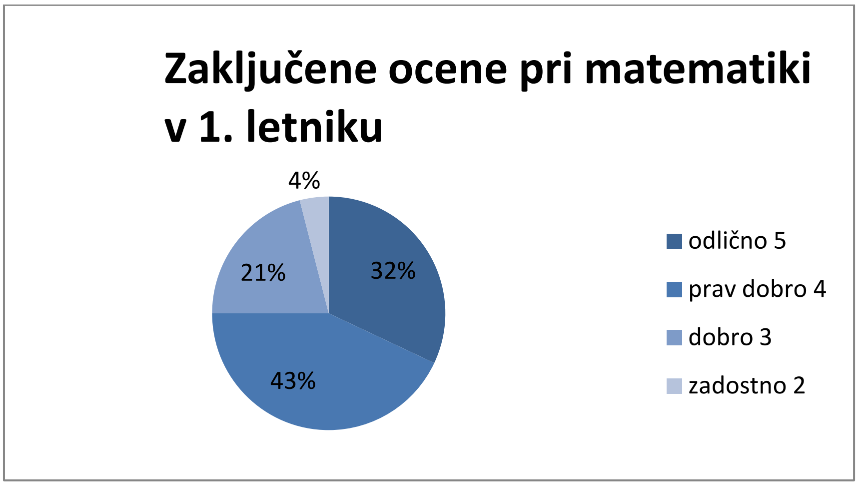
Opomba: Pri prikazu deležev odgovorov na posamezno vprašanje smo za celoto vzeli vse anketirance, razen če je navedeno drugače.

V 1. letniku je bila povprečna zaključena ocena pri matematiki 4,0, število dijakov s posamezno zaključeno oceno pa je zbrano v naslednji tabeli:

Tabela 3: Zaključene ocene anketirancev pri matematiki v 1. letniku

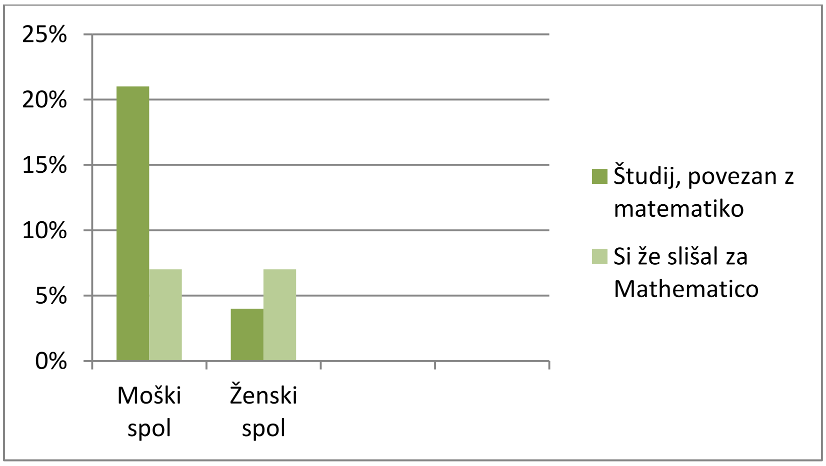
Zaključena ocena pri matematiki	Število dijakov
odlično 5	9
prav dobro 4	12
dobro 3	6
zadostno 2	1

Oglejmo si še tortni diagram, ki prikazuje deleže posameznih zaključenih ocen pri matematiki v 1. letniku.



Graf 25: Zaključene ocene anketirancev pri matematiki v 1. letniku

Z naslednjim grafičnim prikazom bomo predstavili odgovore na vprašanji z zaporednima številkama 3 in 4 (Ali razmišljaš o študiju, povezanem z matematiko? in Si pred današnjo uro matematike že kdaj slišal za program *Mathematica*?) v odvisnosti od spola anketirancev.



Graf 26: Delež anketirancev, ki razmišljajo o študiju matematike in delež anketirancev, ki so že slišali za program *Mathematica,* v odvisnosti od spola

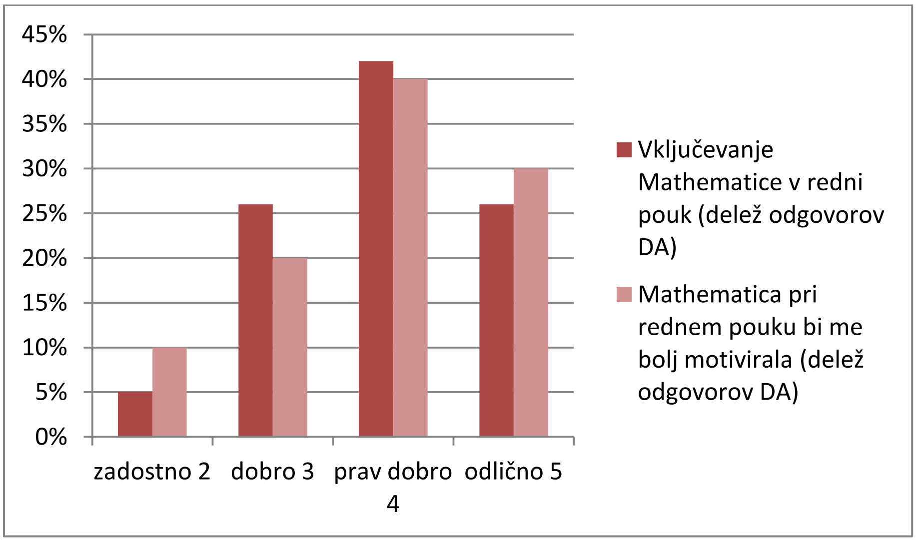
Ugotovili smo, da o študiju, povezanem z matematiko, razmišlja 6 dijakov (21 %) in 1 dijakinja (4 %). Za program *Mathematica* sta že slišala 2 dijaka in 2 dijakinji (7 %). Izmed teh je program uporabljal le 1 dijak.

V nadaljevanju bomo predstavili odgovore na vprašanji 6 in 7, v katerih nas je zanimalo, če bi naj po mnenju anketiranih dijakov program *Mathematica* vključili v redni pouk in če bi delo s tem programom bolj motiviralo dijake pri pouku matematike. S tabelo in stolpičnim diagramom smo prikazali število in vrsto odgovorov glede na zaključeno oceno matematike pri anketiranih dijakih.

Tabela 4: Pregled števila pritrdilnih odgovorov na vprašanji o vključevanju *Mathematice* v redni pouk in povečanju motivacije pri pouku matematike

Zaključena ocena pri matematiki	Vključevanje *Mathematice* v redni pouk (štev. odgovorov DA)	*Mathematica* pri rednem pouku bi me bolj motivirala (štev. odgovorov DA)
odlično 5	5	3
prav dobro 4	8	4
dobro 3	5	2
zadostno 2	1	1

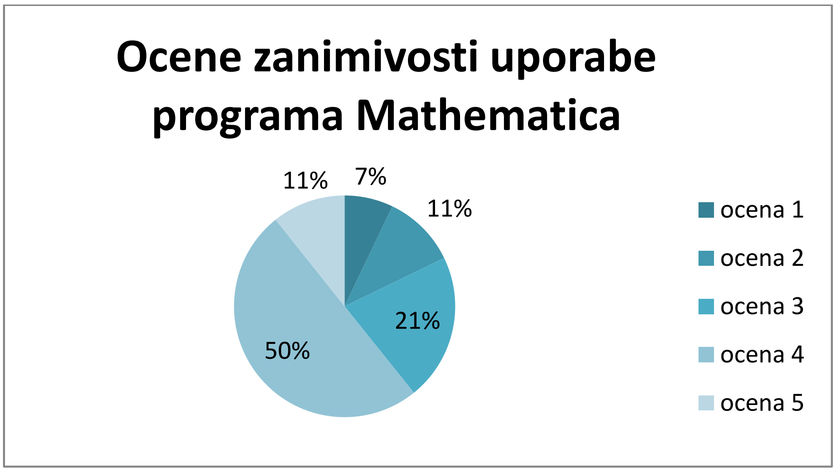
Ugotavljamo, da bi približno 68 % dijakov želelo vključiti *Mathematico* v redni pouk, le približno 36 % dijakov pa bi bilo zaradi tega programa za delo pri pouku matematike bolj motiviranih. V naslednjem grafičnem prikazu smo za celoto vzeli tiste anketirane dijake, ki so na posamezno od obravnavanih vprašanj (vprašanji 6 in 7) odgovorili pritrdilno. Stolpci prikazujejo deleže teh odgovorov glede na zaključene ocene pri matematiki v 1. letniku.



Graf 27: Deleži anketirancev, ki so pritrdilno odgovorili na vprašanje o vključevanju *Mathematice* v redni pouk in o povečanju motivacije pri pouku matematike ob uporabi *Mathematice,* v odvisnosti od zaključene ocene pri matematiki v 1. letniku

Iz grafa razberemo, da največji delež dijakov, ki bi želeli vključitev *Mathematice* v redni pouk, pripada dijakom s prav dobro oceno pri matematiki (42 %). Sledijo dijaki z dobro oziroma odlično oceno (26 %). Ker ima le ena dijakinja oceno zadostno, tega podatka ne bomo vključili v razpravo. Rezultat nas je nekoliko presenetil, saj smo bili mnenja, da bodo prednosti uporabe programa pri pouku najbolje prepoznali dijaki z najvišjimi ocenami pri matematiki. Ugotavljamo tudi, da so v zvezi z dvigom motivacije pri matematiki zaradi vključevanja programa v redni pouk deleži pri dijakih z različnimi ocenami različni: na to vprašanje je pritrdilno odgovorilo 20 % dobrih, 40 % prav dobrih in 30 % odličnih dijakov. Tega rezultata ne znamo pojasniti, pričakovali pa smo, da bo dijakom z nižjimi ocenami pri matematiki uporaba programa dvignila motivacijo pri pouku matematike v večji meri.

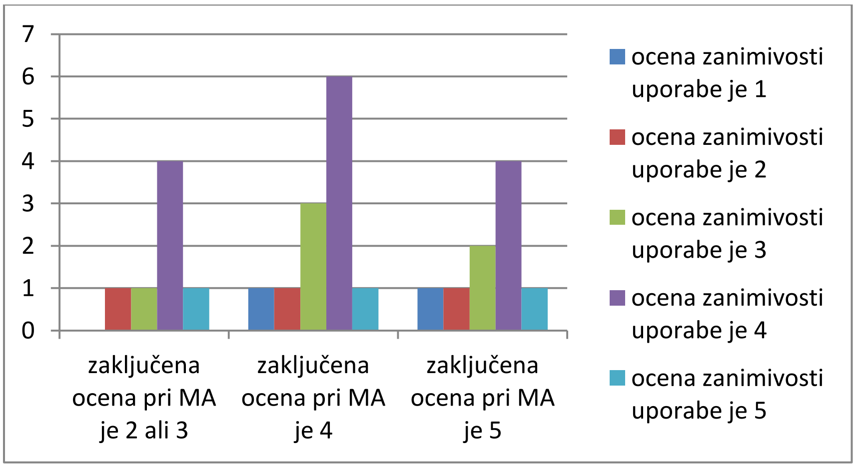
Dijake smo povprašali tudi po oceni zanimivosti uporabe programa *Mathematica* (vprašanje 8). Možne ocene so bile na lestvici od 1 (nezanimivo) do 5 (zelo zanimivo). Odgovore smo prikazali s tortnim diagramom.



Graf 28: Deleži dijakov pri ocenjevanju zanimivosti uporabe programa *Mathematica*

Kar 50 % anketiranih dijakov je ocenilo zanimivost uporabe programa z oceno prav dobro, sledi pa 21 % anketirancev, ki so podali oceno dobro in 11 % anketirancev, ki se jim je zdel program zelo zanimiv. Ugotavljamo, da se le 18 % dijakom zdi uporaba programa *Mathematica* manj zanimiva (oceni 1 in 2). Rezultati so nas razveselili, saj kažejo, da smo s predstavitvijo učne ure vzbudili zanimanje in interes za uporabo *Mathematice* pri reševanju srednješolskih matematičnih nalog.

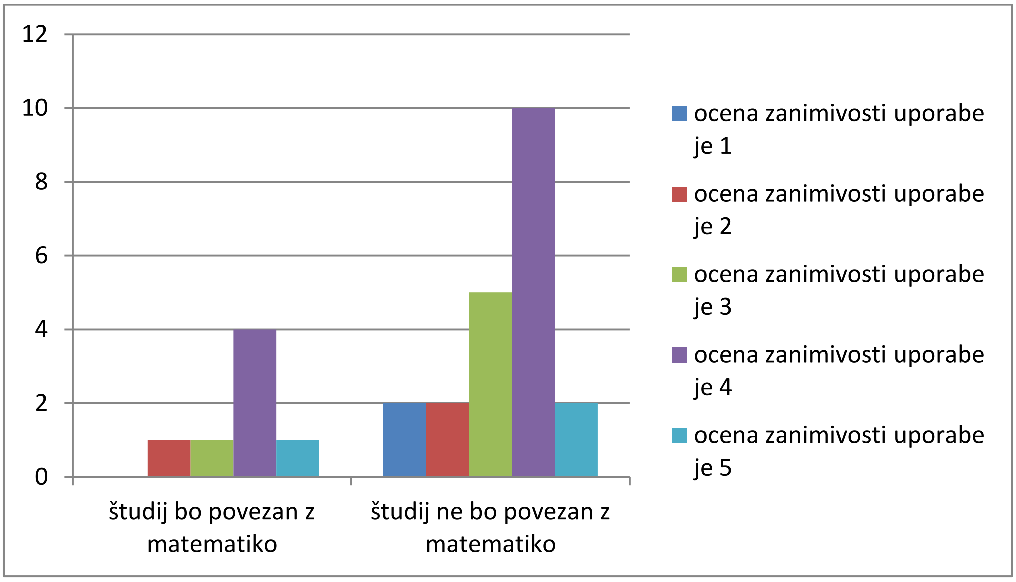
Z namenom, da bomo lahko potrdili ali ovrgli eno izmed zastavljenih hipotez, si oglejmo, katere ocene zanimivosti uporabe programa so izbrali dijaki, ki so imeli v 1. letniku zaključeno oceno pri matematiki 2 ali 3, 4 oziroma 5, prav tako pa nas je zanimalo, katere ocene so izbrali dijaki, ki razmišljajo o študiju, povezanem z matematiko. Rezultate bomo predstavili v naslednjih dveh stolpičnih diagramih.



Graf 29: Število dijakov, ki so ocenjevali zanimivost uporabe programa *Mathematica,* v odvisnosti od zaključene ocene pri matematiki v 1. letniku

Ugotavljamo, da so dijaki zanimivost uporabe programa ocenjevali približno enako v odvisnosti od zaključene ocene pri matematiki v 1. letniku. Slabše ocene (ocena zanimivosti uporabe 1 ali 2) in najboljšo oceno (ocena zanimivosti uporabe 5) je pri vsaki kategoriji zaključene ocene izbralo manjše število dijakov (1 do 2 dijaka). Iz grafa prav tako razberemo, da je pri vsaki kategoriji ocene največ dijakov za oceno zanimivosti uporabe izbralo oceno 4. Izbor ocen je po našem mnenju pričakovan.

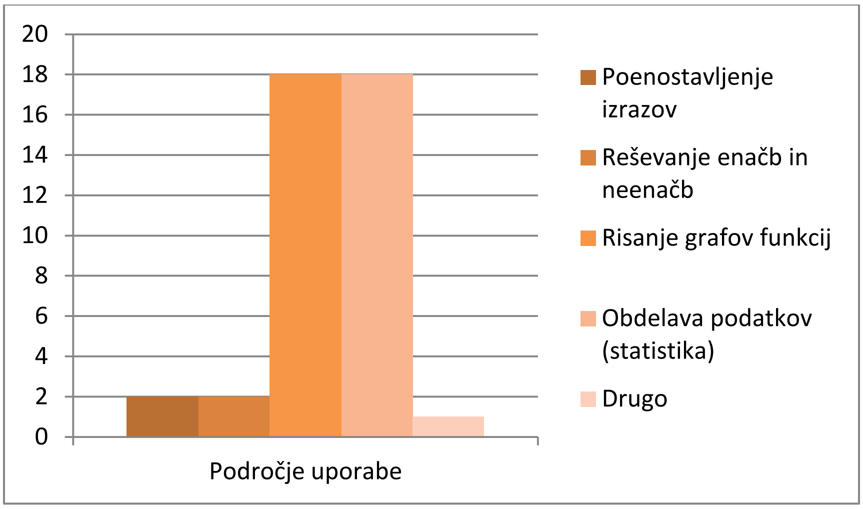
Oglejmo si še prikaz ocen zanimivosti uporabe programa v odvisnosti od razmišljanja dijakov o študiju, povezanem z matematiko.



Graf 30: Število dijakov, ki so ocenjevali zanimivost uporabe programa *Mathematica,* v odvisnosti od želenega študija, povezanega z matematiko

Rezultati kažejo, da ponovno ni bistvenih razlik pri ocenjevanju uporabe zanimivosti programa *Mathematica* pri dijakih, ki bodo verjetno izbrali študij, povezan z matematiko, in tistih, katerih študij ne bo povezan z matematiko. Natančna števila dijakov lahko razberemo iz grafa, sicer pa lahko ugotovimo, da je v obeh kategorijah največ dijakov izbralo oceno prav dobro. Skrajne ocene (1 ali 2 oziroma 5) pa manjše število dijakov (1 do 2 dijaka).

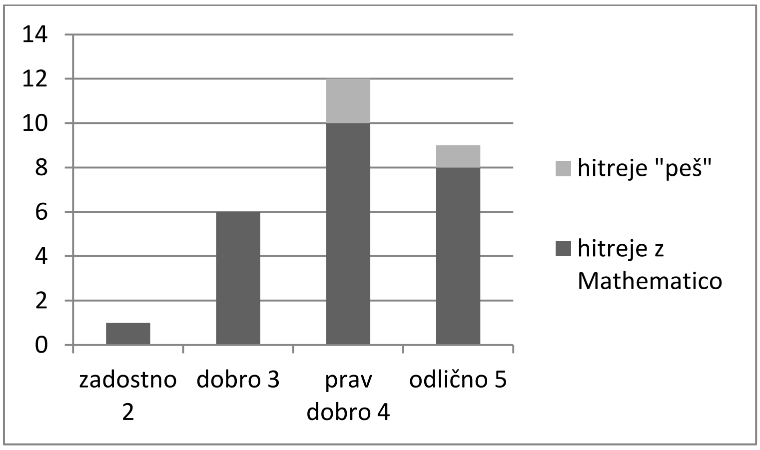
Z vprašanjem številka 9 smo želeli izvedeti, na katerih matematičnih področjih se zdi dijakom program *Mathematica* najbolj uporaben. Med ponujenimi odgovori smo zajeli področja Poenostavljanje izrazov, Reševanje enačb in neenačb, Risanje grafov funkcij in Obdelava podatkov (statistika), dijaki pa so lahko zapisali tudi svoje področje pod rubriko »Drugo«. Ugotovili smo, da je to možnost izbral le en dijak in navedel uporabo programa pri reševanju problemov iz področja strojništva. Sicer so dijaki lahko izbrali dva odgovora. Pri obdelavi podatkov smo vsakemu izbranemu odgovoru podelili eno točko, rezultati so zbrani v naslednjem stolpičnem diagramu.



Graf 31: Število točk posameznega področja, kjer se zdi anketiranim dijakom uporabnost programa *Mathematica* največja

Največ točk (kar 18) sta dobili področji Risanje grafov funkcij in Obdelava podatkov (statistika), sledita pa področji Poenostavljanje izrazov ter Reševanje enačb in neenačb. Glede na izbrane odgovore ugotavljamo, da se zdi dijakom program najbolj uporaben pri reševanju nalog, ki zahtevajo grafične spretnosti in dolgotrajno preštevanje in obdelavo podatkov.

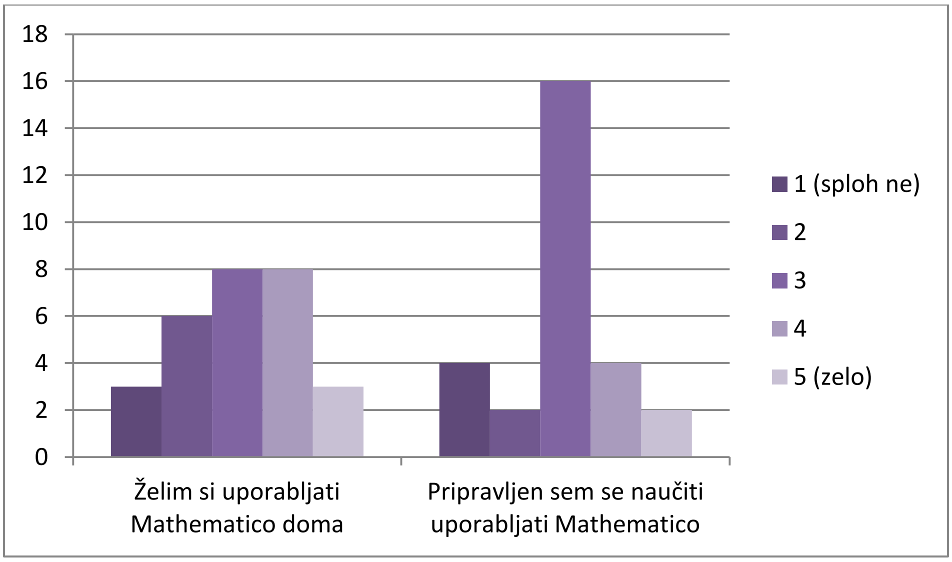
V predstavitvi učne ure s programom *Mathematica* smo prikazali reševanje kompleksne naloge iz področja opisne statistike. Z vprašanjem številka 10 smo želeli izvedeti, ali bi dijaki ob predpostavki poznavanja osnovnih ukazov v programu nalogo rešili hitreje »peš« ali z *Mathematico*. Zbrane rezultate smo predstavili v odvisnosti od zaključenih ocen pri matematiki v 1. letniku. V stolpičnem diagramu smo prikazali števila anketiranih dijakov pri posameznem odgovoru in ne njihovih deležev.



Graf 32: Prikaz števila anketiranih dijakov, ki bi nalogo iz opisne statistike hitreje rešili s programom *Mathematica*, kakor pa »peš«, glede na zaključeno oceno pri matematiki v 1. letniku

Ugotavljamo, da bi velika večina (kar 25 od 28 anketirancev) nalogo iz opisne statistike rešila hitreje z *Mathematico*, kot pa svinčnikom, papirjem in žepnim računalom. Ne preseneča pa, da so trije dijaki, ki ne bi uporabili programa, med tistimi z ocenami prav dobro in odlično.

Ob koncu anketnega vprašalnika nas je zanimalo še, ali bi si dijaki želeli uporabljati program *Mathematica* doma in ali bi se bili pripravljeni naučiti osnovnih ukazov programa (vprašanji 11 in 12). Dijaki so odgovore izbrali na lestvici ocen od 1 (sploh ne) do 5 (zelo). Število anketirancev, ki so se odločili za posamezno oceno, smo za obe vprašanji predstavili v istem grafu – posamezen stolpični diagram prikazuje odgovore na eno vprašanje.



Graf 33: Prikaz števila anketirancev, ki so izbirali med ocenami od 1 (sploh ne) do 5 (zelo), pri želji po uporabi programa *Mathematica* doma in pripravljenosti naučiti se uporabljati program *Mathematica*

Iz prikazanih podatkov razberemo, da je na vprašanje o želji uporabljati program doma 16 dijakov (več kot polovica) izbrala oceni 3 in 4, po trije dijaki pa si sploh ne želijo oziroma zelo želijo uporabljati program doma. Pri odgovoru na vprašanje o pripravljenosti naučiti se ukazov za uporabo programa je 16 dijakov pokazalo srednjo pripravljenost (ocena 3), po 6 dijakov pa večjo (oceni 4 in 5) oziroma manjšo pripravljenost (oceni 1 in 2). Rezultati kažejo naklonjenost anketiranih dijakov programu *Mathematica*. Če pri tej naklonjenosti upoštevamo izbrane ocene 3, 4, in 5, ugotavljamo, da si večina dijakov želi uporabljati program doma in da so kljub mnogim drugim šolskim in obšolskim dejavnostim pripravljeni nameniti svoj prosti čas za učenje programa. Menimo, da bi bil interes za *Mathematico* še večji, če bi dijaki tudi sami preizkusili uporabnost programa.

7. RAZPRAVA

V tem poglavju bomo na osnovi raziskovanja posameznih ukazov v programu *Mathematica* in na osnovi rezultatov ankete odgovorili na zastavljena raziskovalna vprašanja ter potrdili ali ovrgli hipoteze, ki smo jih zapisali v poglavjih 1.3 in 1.4. Pri tem se zavedamo, da je bilo število anketiranih dijakov zaradi potrebne predhodne praktične izvedbe učne ure majhno (28 anketirancev) in tako analize rezultatov anketiranja in s tem povezanega (ne)potrjevanja zastavljenih hipotez ne moremo posploševati na širše skupine srednješolcev.

Raziskovalna vprašanja

Ali lahko dijak sam le ob uporabi obstoječe literature (brez uvajalnega tečaja) usvoji uporabo osnovnih ukazov v programu Mathematica?

Na to vprašanje bomo odgovorili pritrdilno. Znotraj programa *Mathematica* obstaja izredno izčrpna »pomoč«, ki je enostavno dostopna. Vsi ukazi so razloženi splošno (teoretično), nato pa uporabljeni na mnogih primerih. Razen tega so navedene še dodatne, kombinirane in kompleksnejše možnosti uporabe posameznega ukaza. Obstaja tudi nekaj tiskanih virov (na primer vira [1] in [2]), s katerimi si lahko učinkovito pomagamo pri reševanju problemov z *Mathematico*. Ker sva se avtorja raziskovalne naloge sama »dokopala« do uporabljenih programskih ukazov, meniva, da bi bili pri tem uspešni tudi najini vrstniki, ki bi si *Mathematico* zares želeli uporabljati in bili dovolj pogumni in vztrajni.

Ali lahko naloge iz večine matematičnih vsebin 1. in 2. letnika gimnazijskega programa rešimo s programom Mathematica?

Tudi na to vprašanje bomo odgovorili pritrdilno. Seveda pa se moramo pri tem zavedati, da programu ne moremo le podati besedila naloge in pričakovati ustrezne rešitve. Pri reševanju nalog smo uspešni le, če izkoristimo svoje matematično znanje ter izberemo pravilne programske ukaze in upoštevamo potrebno sintakso. *Mathematica* lahko zelo hitro poenostavi izraz, reši enačbo, natančno nariše graf funkcije, v kratkem času obdela in predstavi množico podatkov, ne more pa razmišljati namesto nas. Izvrši le ukaze, ki jih poda njen uporabnik.

Ali bodo dijaki pozitivno sprejeli uporabo programa Mathematica pri pouku in se ga želeli naučiti uporabljati?

Na to raziskovalno vprašanje bomo odgovorili na osnovi rezultatov anketnega vprašalnika. Dijaki so z ocenami od 1 (sploh ne) do 5 (zelo) ocenili zanimivost uporabe programa *Mathematica* pri učni uri. Rezultati ankete kažejo, da se le 18 % dijakov zdi uporaba programa *Mathematica* manj zanimiva (oceni 1 in 2). Kar 61 % dijakov je zanimivost uporabe programa ocenilo s 4 oziroma s 5. Ugotavljamo, da je večina dijakov spoznala predstavljen program kot zanimiv in uporaben. Menimo, da bi bil delež nad programom navdušenih dijakov še večji, če bi predstavitev zajemala več različnih področjih uporabe programa in če bi dijaki lahko tudi sami preizkusili uporabno moč *Mathematice*.

<u>Hipoteze</u>

Ob ustreznem matematičnem znanju in zadostnem poznavanju angleškega jezika se dijak lahko sam nauči uporabe osnovnih ukazov v programu Mathematica.

Hipotezo lahko potrdimo. Razlogi so navedeni v odgovoru na prvo raziskovalno vprašanje.

S programom Mathematica lahko rešimo mnoge naloge iz različnih matematičnih vsebin v 1. in 2. letniku gimnazijskega programa.

Tudi to hipotezo lahko potrdimo. Utemeljitev je ista kot pri odgovoru na drugo raziskovalno vprašanje.

Ob osnovnem predznanju uporabe programa nalogo z Mathematico rešimo hitreje, kakor če bi jo reševali 'peš.'

Hipotezo potrdimo na osnovi analize odgovorov pri anketnem vprašanju številka 10. Ugotavljamo, da kar 89 % anketiranih dijakov meni, da bi ob poznavanju osnovnih ukazov v *Mathematici* nalogo iz opisne statistike rešili hitreje z uporabo programa kot pa »peš«.

Program bi pritegnil predvsem tiste dijake, ki so na področju matematike uspešni ali razmišljajo o študiju, povezanem z matematiko.
Hipotezo zavrnemo na osnovi analize odgovorov na anketna vprašanja 2, 3 in 8. Ugotavljamo, da ni bistvenih razlik med ocenjevanjem zanimivosti uporabe programa *Mathematica* med dijaki z različnimi zaključenimi ocenami pri matematiki v 1. letniku ali med dijaki, ki razmišljajo o študiju, povezanem z matematiko in tistimi, ki o takem študiju ne razmišljajo. Podrobnejša utemeljitev zavrnitve hipoteze je podana z grafoma 29 in 30 ter njunima analizama.

Če bi imeli možnost, bi dijaki uporabo programa vključili v redni pouk.
Hipotezo potrdimo na osnovi odgovorov anketnega vprašanja številka 6, saj bi 68 % anketiranih dijakov delo s programom *Mathematica* vključilo v redni pouk.

Uporaba programa pri pouku bi povečala motiviranost dijakov za delo.
Hipotezo zavrnemo. Ugotovili smo, da bi le približno 36 % anketiranih dijakov bilo zaradi uporabe programa *Mathematica* za delo pri pouku matematike bolj motiviranih.

Dijaki nimajo interesa za uporabo programa Mathematica doma.
Dijaki se niso pripravljeni naučiti osnov uporabe programa Mathematica.
Izkaže se, da obeh zadnjih zapisanih hipotez ne moremo ne potrditi, ne zavrniti. Izračunali smo namreč povprečni oceni za lestvico ocenjevanja od 1 (sploh ne) do 5 (zelo). Pri vprašanju o želji po uporabi *Mathematice* doma je povprečna ocena 3,1, pri vprašanju o pripravljenosti naučiti se uporabljati *Mathematico* pa 2,9. Povprečni oceni sta torej približno na sredi ocenjevalne lestvice.

8. ZAKLJUČEK

Glavni del raziskovalne naloge predstavlja proučevanje računalniškega programa *Mathematica* z namenom ugotoviti, ali lahko dijak splošnega gimnazijskega programa samostojno razišče tiste programske ukaze, ki so potrebni za reševanje problemov srednješolske matematike. Zaradi dostopnosti ustrezne literature in »pomoči uporabniku«

znotraj programa raziskovanje programskih ukazov ni predstavljalo večjih težav. Pri tem smo prišli do spoznanja, da je *Mathematica* koristen pripomoček pri reševanju matematičnih nalog, predstavljenih v učbenikih za 1. in 2. letnik gimnazijskega programa. Pripomniti želimo, da lahko uporabimo program le za preverjanje rezultatov ali na primer za prikaz natančnejšega risanja grafov funkcij in ne za nadomestilo našega učenja matematične snovi.

Razen samega programa *Mathematica* smo raziskali tudi odziv dijakov na predstavitev učne ure iz področja opisne statistike, ki je temeljila na uporabi *Mathematice.* Analiza odgovorov na anketna vprašanja je pokazala, da je program pritegnil dijake z različnim matematičnim znanjem in da si večina anketiranih dijakov želi vključevanja *Mathematice* v redni pouk. Presenetila nas je ugotovitev, da uporaba *Mathematice* pri večini anketiranih dijakov ne bi dvignila motivacije pri rednem pouku matematike. Spet pa je želja po uporabi *Mathematice* doma in želja naučiti se uporabljati *Mathematico* večja od pričakovane.

V sklepnem delu raziskovalne naloge bomo nakazali še možnosti, ki so osnova za nadaljnje raziskovalno delo. V našem primeru bi navedli dve področji raziskovanja:

1. proučevanje ukazov *v Mathematici*, ki omogočajo reševanje matematičnih problemov iz vsebin druge polovice 2. letnika ter 3. in 4. letnika splošnega gimnazijskega programa. Na ta način bi dobili izčrpen seznam ukazov za reševanje nalog srednješolske matematike, mnogi izmed teh ukazov pa bi bili koristni tudi na področjih visokošolskega izobraževanja, ki so kakorkoli povezana z matematiko.
2. raziskovanje možnosti uporabe *Mathematice* na naravoslovnih področjih (biologija, kemija, fizika). Pri tem imamo v mislih reševanje različnih naravoslovnih problemov kot tudi uporabo že vgrajenih aplikacij.

Avtorja raziskovalne naloge bi na osnovi proučevanja programa *Mathematica* izpostavila naslednje pozitivne lastnosti:

- možnost uporabe na različnih predmetnih področjih
- velika hitrost pridobivanja (skoraj poljubno) natančnih rezultatov
- estetsko narisani grafi funkcij
- dinamično spreminjanje odvisnih količin
- možnosti uporabnikovih nastavitev
- obširna in nazorna pomoč uporabniku s številnimi ponazoritvenimi primeri

Zaradi navedenega meniva, da bi bilo na najini srednji šoli zelo koristno, če bi v računalniški učilnici namestili program *Mathematica*. Prepričana sva, da bi ga za popestritev pri pouku in mnoge nazornejše predstavitve obravnavane snovi uporabljali profesorji matematike in tudi profesorji drugih (predvsem naravoslovnih) predmetov. Seveda pa bi ga lahko koristno uporabljali tudi dijaki. V okviru raziskovalne naloge je med drugim nastal zajeten seznam ukazov v programu *Mathematica*, ki je lahko v pomoč profesorjem in dijakom. Večina ukazov je razloženih in uporabljenih v konkretnih primerih reševanja nalog srednješolske matematike.

Morda bo najina raziskovalna naloga spodbuda kakšnemu novemu uporabniku *Mathematice*.

9. LITERATURA IN VIRI

Literatura

[1] M. Gašperšič: *Mathematica do bruhanja*, samozaložba, Trzin, 2010.

[2] M. Gašperšič: *Mathematica do bruhanja II*, samozaložba, Trzin, 2010.

[3] G. Pavlič, D. Kavka, M. Rugelj, J. Šparovec: Linea nova, Matematika za gimnazije, Modrijan založba, 1. izd., Ljubljana, 2011.

[4] G. Pavlič, D. Kavka, M. Rugelj, J. Šparovec: Planum novum, Matematika za gimnazije, Modrijan založba, 1. izd., Ljubljana, 2012.

Viri

[5] O *Mathematici* v Wikipediji [Online]

Dostopno na URL naslovu: http://en.wikipedia.org/wiki/Mathematica (povzeto 15. 9. 2012 ob 10.45)

[6] Osnovno o programu *Mathematica* [Online]

Dostopno na URL naslovu: http://www.wolfram.com/mathematica/ (povzeto 13. 9. 2012 ob 16.00)

[7] M. Petkovšek, B. Plestenjak: Osnovno o *Mathematici* [Online]

Dostopno na URL naslovu: http://atom.uni-mb.si/~UKEMAT/OsnovnaNavodilaMMa.pdf (povzeto 13. 9. 2012 ob 15.30)

[8] Pregled osnovnih funkcij v programu *Mathematica* [Online]

Dostopno na URL naslovu:

http://reference.wolfram.com/legacy/calccenter/Functions/BasicMath/BasicFunctions/index.en.html (povzeto 10. 9. 2012 ob 17.40)

[9] Pregled verzij programa *Mathematica* [Online]

Dostopno na URL naslovu:

http://www.wolfram.com/mathematica/quick-revision-history.html (povzeto 25. 1. 2013 ob 17.20)

[10] WolframAlpha blog [Online]

Dostopno na URL naslovu: http://blog.wolframalpha.com/ (povzeto 7. 11. 2012 ob 19.00)

Viri slik

Slika 10: Osnutek programa *Mathematica* leta 1986 [Online]

Dostopno na URL naslovu:
http://www.wolfram.com/company/scrapbook/page02.html#preWRI_InventingTheLanguage
(povzeto 13. 9. 2012 ob 16.10)

Slika 2: Verzija 1.0 leta 1988 na računalniku Macintosh [Online]

Dostopno na URL naslovu:

http://www.wolfram.com/company/scrapbook/page03.html#1988_StylishMathematicaStartup

(povzeto 13. 9. 2012 ob 16.20)

Slika 3: Verzije programa *Mathematica* [Online]

Dostopno na URL naslovu:

http://www.wolfram.com/company/scrapbook/page09.html#2003_MathematicaBook

(povzeto 13. 9. 2012 ob 16.15)

Slika 4: Trirazsežni geometrijski objekt [Online]

Dostopno na URL naslovu:

http://demonstrations.wolfram.com/RhombicHexecontahedronTower/

(povzeto 13. 9. 2012 ob 16.30)

Slika 5: Diagram CIE barvnega prostora [Online]

Dostopno na URL naslovu: http://demonstrations.wolfram.com/CIEChromaticityDiagram/

(povzeto 13. 9. 2012 ob 16.45)

Slika 6: Zemljevid angleško govorečih območij po svetu [Online]
Dostopno na URL naslovu: http://demonstrations.wolfram.com/CountryGroups/
(povzeto 13. 9. 2012 ob 16.50)

Slika 7: Prikaz molekule etana [Online]
Dostopno na URL naslovu:
http://demonstrations.wolfram.com/RotationAboutCarbonCarbonBonds/
(povzeto 13. 9. 2012 ob 17.05)

Slika 8: Očesno zrklo z mišicama obračalkama [Online]
Dostopno na URL naslovu: http://demonstrations.wolfram.com/EyeballAndTwoMuscles/
(povzeto 13. 9. 2012 ob 17.10)

Slika 9: Mehanični zglob [Online]
Dostopno na URL naslovu: http://demonstrations.wolfram.com/UniversalJoint/
(povzeto 13. 9. 2012 ob 17.20)

ZAPIS O DRUŽBENI ODGOVORNOSTI

V skladu z definicijo in načeli družbene odgovornosti najina raziskovalna naloga prispeva k družbeno odgovornemu ravnanju do drugih posameznikov in do narave na naslednji način:

Raziskovalna naloga naj bo v digitalni obliki dostopna vsem zainteresiranim posameznikom. S tem omogočava uporabo raziskovalnih izsledkov z namenom širjenja obstoječega znanja in pridobivanja novih spoznanj na različnih področjih. Ker natisnjene naloge ne bo potrebno razmnoževati (fotokopirati), bomo varčevali tudi s papirjem in tako prispevali k ohranitvi narave.

Avtorja