PDF kalkulator

Šarić, Slađana

Master's thesis / Diplomski rad

2015

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:034736

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-24**

Repository / Repozitorij:

Repository of the Faculty of Science - University of Zagreb

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Slađana Šarić

PDF KALKULATOR

Diplomski rad

Voditelj rada:
izv. prof. dr. sc. Jadranka Mićić Hot

Zagreb, studeni 2015.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Sadržaj

Sadržaj **iii**

Uvod **1**

1 PDF kalkulator **2**
1.1 Uvod 2
1.2 Paket *insDLJS* 14
1.3 O eksponencijalnim funkcijama 18
1.4 Primjeri korištenja paketa *eforms* i *insDLJS* 24

2 Paket eDucation Bundle **40**
2.1 Paket eDucation Bundle 40
2.2 Paket *exerquiz* 41
2.3 Osnovne opcije paketa 42
2.4 Okolina *exercise* 44
2.5 Okolina *shortquiz* 49
2.6 Okolina *quiz* 61
2.7 Pitanja objektivnog tipa 65
2.8 Proširenje s *dljslib* i *insdljs* 73
2.9 Postavljanje kviza na web server 74
2.10 Primjena Acrobat eDucation Bundlea 77

Bibliografija **95**
Literatura 95

Uvod

U ovom diplomskom radu će biti obrađena dva paketa iz AcroTeX softwarea: kalkulator i kviz. Izlazni dokument za oba paketa je PDF. S tim paketima možemo kreirati kalkulatore gdje korisnici unose željene parametre u PDF i klikom na određeni gumb dobiju rješenje. Drugi paket služi za kreiranje raznih kvizova, ispita te vježbi s rješenjima ili bez. Taj paket je može biti od velike pomoći nastavnicima, pogotovo onima koji imaju puno učenika. Kod sastavljanja ispita mogu se uključiti i JavaScript funkcije pa one ispravljaju i ocjenjuju ispite. Dobra stvar (ili za neke loša) za učenike je što na točnost odgovora ne utječe nastavnikova subjektivnost.
"AcroTeX" se ustvari sastoji od 2 riječi: Acrobat i LaTeXi u njemu se kombiniraju najbolje osobine od oba softwarea. LaTeXpaketi s Adobe Acrobatom pružaju mogućnost kreiranja PDF dokumenata s puno lijepih osobina.
Osim objašnjenja rada s tim paketima, obrađene su i 2 teme iz srednjoškolske matematike: derivacije i eksponencijalne funkcije te sastavljeni kvizovi i kalkulatori na te teme.

Poglavlje 1

PDF kalkulator

1.1 Uvod

Pdf kalkulator omogućava korisniku unos vrijednosti parametara u za to predviđena polja u Acrobat dokumentu. Akcija računanja se pokreće pritiskom na gumb te Acrobat dokument ispiše rezultat tražene operacije. Navedena funkcionalnost postiže se korištenjem paketa `eforms` i `insdljs`.

Paket *eforms*

Paket `eforms` ima sljedeće opcije:

- `dvipsone`
- `dvips`
- `pdftex`
- `dvipdfm`

Svaka opcija ima svoj poseban kôd koji se treba koristiti u kreiranju polja na formi. Tako je, na primjer, naredba za korištenje paketa za `pdftex` ova:
`\usepackage[pdftex]{eforms}`.

Elementi Acrobat formi

Paket `eforms` omogućava korištenje sljedećih elemata Acrobat formi:

- Elementi koji spadaju u kategoriju gumba

  - Push Button
  - Check Box
  - Radio

- Elementi koji spadaju u kategoriju polja za odabir
  - List Box

- Elementi za tekstualni unos
  - Text Field

- Elementi za potpisivanje
  - Signature Field –polje može biti potpisano koristeći Acrobat ili programski iz Latex-a

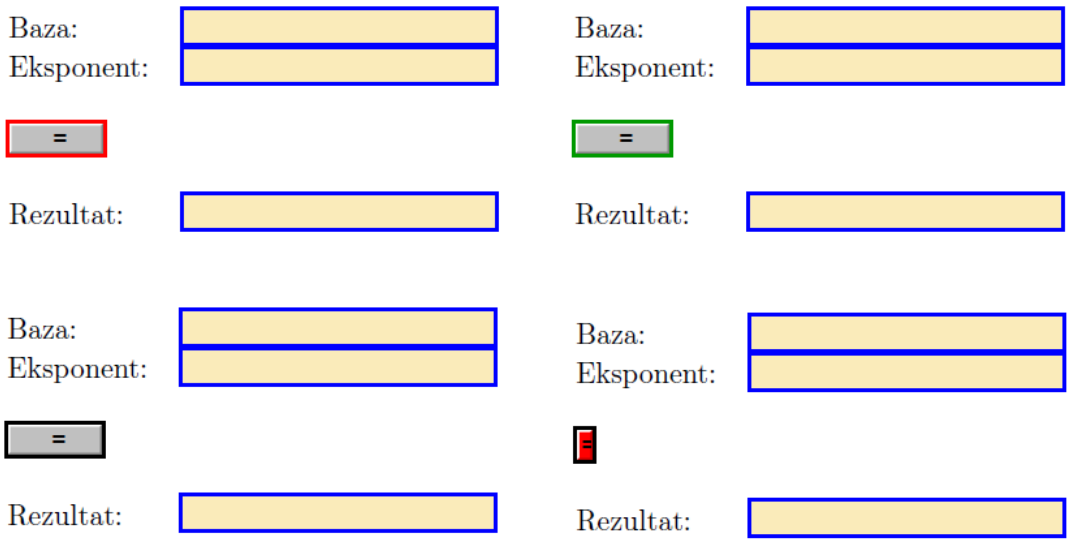
Push Button

Element `push button` je element koji nema vrijednost. Koristi se za iniciranje akcija kao što su pozivi JavaScript funkcija. Navedeni element se može definirati kroz 4 parametra.
`\pushButton[#1]{#2}{#3}{#4}`
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,
- #4: visina pravokutnika u kojem se iscrtava (renderira) element.

Zadani (defaultni) izgled za navedeni element je određen sljedećim parametrima:
`\W{1}\S{B}\F{\FPrint}\BC{0 0 0}\H{P}\BG{.7529 .7529 .7529}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

Na sljedećoj slici su prikazane 4 različite tipke za znak jednakosti.



Check Box

Element `check box` je element koji ima jednu od dvije vrijednosti (uključeno ili isključeno). Korisnik može definirati naziv stanja uključeno, dok se za naziv stanja isključeno koristi vrijednost „Off“.
Navedeni element se može definirati kroz 5 parametra.
`\checkBox[#1]{#2}{#3}{#4}{#5}`
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,
- #4: visina pravokutnika u kojem se iscrtava (renderira) element,
- #5: naziv vrijednosti koja predstavlja uključeno stanje.

Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\W{1}\S{S}\BC{0 0 0}\F{\FPrint}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

Radio Button

Element `radio button` ima slično ponašanje kao i element `check box` samo što se najčešće koristi u uniji s drugim elementima istog tipa. Navedeni element se može definirati kroz 5 parametra.
`\radioButton[#1]{#2}{#3}{#4}{#5}`.
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,
- #4: visina pravokutnika u kojem se iscrtava (renderira) element,
- #5: naziv vrijednosti koja predstavlja uključeno stanje.

Skup `radio button` instanci koje se koriste zajedno trebaju imao isti naziv (parametar #2) no moraju imati drugačije nazive za uključena stanja (parametar #5). Navedeni element se može koristiti u slučajevima kada korisnik može odabrati samo jedan odgovor od više ponuđenih.
Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\W{1}\S{S}\BC{0 0 0}\F{\FPrint}`
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

List box

Element `list box` je element koji sadrži tekstualne stavke pri čemu samo jedna stavka može biti odabrana. Prilikom prikaza navedenog dokumenta vidljiv je određen broj stavki koji stane u pravokutnik u kojem se iscrtava element. Ukoliko postoji više stavki pojavljuje se traka za navigaciju (scroll bar).
Navedeni element se može definirati kroz 5 parametra:
`\listBox[#1]{#2}{#3}{#4}{#5}`.
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,

- #4: visina pravokutnika u kojem se iscrtava (renderira) element,
- #5: skup vrijednosti koje se prikazuju.

Vrijednost parametra #5 koji definira skup vrijednosti koje se prikazuju može se upisati u 2 različita oblika:

- Kao polje vrijednosti prije čemu se svaka vrijednost sastoji od dva parametra `[(v1)(item1)][(v2)(item2)]...[(vn)(itemn)]`.
  Vrijednost prvog parametra označava vrijednost koji će izlazna varijabla poprimiti kada se odabere zapis dok se vrijednost drugog parametra prikazuje korisniku.
- Kao polje znakovnih nizova (stringova) `(item1) (item2)]...(item)`
  U tom slučaju vrijednost izlazne varijable će biti jednaka vrijednosti koja se prikazuje korisniku.

Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\W{1}\S{I}\F{\FPrint}\BC{0 0 0}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

Combo Box

Element `combo box` je element koji sadrži tekstualne stavke koje se prikazuju u padajućoj listi. Element ima svojstvo da se na kraj liste mogu dodavati nove stavke.
Navedeni element se može definirati kroz 5 parametra:
`\listBox[#1]{#2}{#3}{#4}{#5}`.
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,
- #4: visina pravokutnika u kojem se iscrtava (renderira) element,
- #5: skup vrijednosti koje se prikazuju.

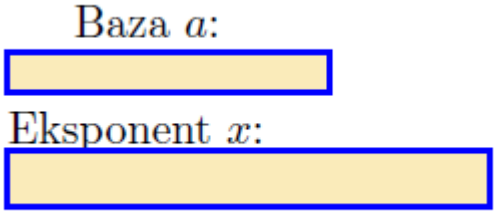
Mogući načini zadavanja parametra #5 su identični kao i kod elementa `list Box`.
Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\W{1}\S{I}\F{\FPrint}\BC{0 0 0}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

Text field

Element `text field` služi za unos teksta u formu.
Navedeni element se može definirati kroz 4 parametra:
`\textField[#1]{#2}{#3}{#4}`.
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti,
- #2: proizvoljan naziv za instancu elementa,
- #3: širina pravokutnika u kojem se iscrtava (renderira) element,
- #4: visina pravokutnika u kojem se iscrtava (renderira) element.

Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\F{\FPrint}\BC{0 0 0}\W{1}\S{S}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.
Jedna od promjena je vidljiva na ovoj slici:



Signature Field

Element `signature field` jest element koji može biti digitalno potpisan.
Navedeni element se može definirati kroz 4 parametra:
`\sigField[#1]{#2}{#3}{#4}`.
Opis parametara:

- #1: opcijski parametar koji se koristi za bilo kakvu izmjenu u izgledu koja odstupna od postavljenih (default) vrijednosti;
- #2: proizvoljan naziv za instancu elementa;
- #3: širina pravokutnika u kojem se iscrtava (renderira) element;

- #4: visina pravokutnika u kojem se iscrtava (renderira) element.

Zadani (default) izgled za navedeni element je određen sljedećim parametrima:
`\F{\FPrint}\BC{}\BG{}\W{1}\S{S}`.
Značenje korištenih varijabli je opisano u poglavlju s popisom važnijih varijabli.

Akcije

Osim što polja na formi mogu služiti za skupljanje podataka od korisnika, mogu i izvršavati jednu ili više akcija. Akcije uključuju izvršavanje JavaScript kôda, odlazak na pojedinu stranicu na dokumentu, otvaranje datoteke, resetiranje forme i slično. Počevši s Acrobat verzijom 5.0 većina akcija se može izvesti koristeći JavaScript metode.
Akciju pokreću okidači (trigger). Jedno polje na formi može imati više akcije pri čemu se svaka od akcija pokreće posebinom okidačem.
Tipovi okidača:

- Mouse enter – okidač se okida kada se mišem dođe u područje koje definira pravokutnik u kojem se nalazi polje. `\AAMouseEnter` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
  Primjer kôda:

  ```
  \textField[\AA{\AAMouseEnter{%
  \JS{app.alert("Ušli ste u polje")}}}]
  {polje}{2in}{15bp}
  ```

- Mouse exit - okidač se okida kada se mišem izađe iz područja koje definira pravokutnik u kojem se nalazi polje. `\AAMouseExit` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.

- Mouse down - okidač se okida kada je pritisnuta lijeva tipka miša unutar područja koje definira pravokutnik u kojem se nalazi polje. `\AAMouseDown` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.

- Mouse up - okidač se okida kada je otpuštena lijeva tipka miša unutar područja koje definira pravokutnik u kojem se nalazi polje. `\AAMouseUp` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.

- On Focus - okidač se okida kada se polje nađe u fokusu i to na bilo koji način. Polje se može naći u fokusu ukoliko se lijevom tipkom miša klikne u polje ili se tab tipkom dođe u polje s kursorom. `\AAOnFocus` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.

- On Blur - okidač se okida kada polje izgubi fokus. Polje može izgubiti fokus ukoliko se lijevom tipkom miša klikne negdje van polja ili se tab tipkom izađe iz polja. `\AAOnBlur` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- Format - okidač se okida kada se tekst unese u tekstualno polje. Opcionalno, može se definirati funkcija u JavaScript kôdu koja će onda formatirati upisani tekst. `\AAFormat` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- Keystroke - okidač se okida svaki put kada se unese bilo koji znak u tekstualno polje. Opcionalno, može se definirati funkcija u JavaScript kôdu koja će onda obraditi uneseni znak ili znakove. `\AAKeystroke` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- Validate - okidač za koji se može definirati JavaScript kôd koji može odraditi validaciju podataka. `\AAValidate` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- Calculate - okidač za koji se može definirati JavaScript kôd koji može odraditi određena računanja na temelju unesenih podataka u jedno ili više polja na formi. `\AACalculate` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- PageOpen - okidač za koji se može definirati akcija koja će se izvršiti nakon što se otvoriti stranica (npr. kada korisnik dođe na stranicu preko linka). `\AAPageOpen` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- PageClose - okidač za koji se može definirati akcija koja će se izvršiti prije nego što se zatvoriti stranica (npr. otići će se na neku drugu stranicu u dokumentu). `\AAPageClose` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- PageVisible - okidač za koji se može definirati akcija koja će se izvršiti kada stranica postane vidljiva u pregledniku. `\AAPageVisible` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.
- PageInvisible - okidač za koji se može definirati akcija koja će se izvršiti kada stranica više nije vidljiva u pregledniku. `\AAPageInvisible` ključ se koristi unutar `\AA` argumenta za definiranje ovog tipa okidača.

Tipovi akcija:

- GoTo – odlazak na destinaciju u trenutnom dokumentu;
- GoToR – odlazak na destinaciju u drugom dokumentu;
- Launch – otvaranje datoteke;

- URI – otvaranje linka na webu;
- Named – pokretanje akcija koji su unaprijed definirane od strane Acrobat preglednika. Primjeri za takve akcije su sljedeća stranica (NextPage), prethodna stranica (PrevPage), prva stranica (FirstPage) i zadnja stranica (LastPage);
- SubmitForm – slanje podataka s forme na određenu lokaciju;
- JavaScript – izvršavanje JavaScript akcija.

Popis važnijih podržanih varijabli koje se koriste za definiranje svojstva elemenata unutar forme

Naredba	**Opis**	**Primjer**

`\F`	Određuje vidljivost
	atributa unutar forme.
	Može sadržavati vrijednosti
	`\FHidden, \FPrint,`
	`\FNoView, \FLock`
		`\F{}`

`\W` `\W`	Širina pravokutnika u kojem se element nalazi	`\W{}` što je isto kao i `\W{1}`
`\S`	Stil linije pravokutnika u kojem se element nalazi. Moguće su vrijednosti S (solid), D (dashed), B (beveled), I (inset), U (underlined). Primjer za način apliciranja pojedinog stila je `\S{B}`	`\S{}`
`\AA`	Dodatne akcije koje se izvršavaju na određene okidače kao što su pristisak tipke miša, izlazak kursora iz polja i slično	`\AA{}` (nema akcija)
`\A`	Koristi se za definiranje Javascript akcija	`\A{}` nema akcija
`\Border`	Definira na koji se način označava granica. Sastoji se od tri brojčana elementa. Ako su svi elementi 0 granica se ne iscrtava	`\Border{0 0 0}`
`\AP`	Koristi se kod elemenata tipa check box za definiranje vrijednosti uključeno	`\AP{}`
`\AS`	Stanje za pojavljivanje koje se koristi kod elemenata tipa check box i radio button kada postoji više od jednog načina pojavljivanja. Koristi se za napredne tehnike.	`\AS{}`
`\TU`	Koristi se za definiranje objašnjenja polja (tooltip). Npr. `\TU{Ulica}`	`\TU{}`

`\Ff`	Definira razne parametre koje utječu na sam izgled i ponašanje elemenata. Primjeri parametara: `\FfReadOnly, \FfRequired, \FfNoExport, \FfMultiline, \FfPassword, \FfNoToggleToOff, \FfRadio, \FfPushButton`. Razni parametri se mogu primjeniti samo na pojedine tipove elemenata.	`Ff{}`
`\DV`	Zadana (default) vrijednost polja	`\DV{}`
`\V`	Trenutna vrijednost polja	`\V{}`
`\H`	Definira naglašavanje polja. Definira naglašavanje polja. Moguće vrijednosti su: N (none), P (push), O (outline), I (invert)	`\H{}` što je isto kao i `\H{I}`
`\R`	Broj stupnjeva za koji je element rotiran u suprotnom smjeru od kazaljke na satu. Vrijednost treba biti višekratnik od 90. Npr. `\R{90}`	`\R{}`
`\BC`	Boja ruba elementa. Vrijednost je lista od 0 (transparentno), 1 (sivo), 3 (RGB paleta), 4 (CMYK paleta) brojeva između 0 i 1. Npr. `\BC{1 0 0}` predstavlja crveni rub.	`\BC{}` (transparentno)
`\BG`	Boja pozadine. Vrijednost je lista od 0 (transparentno), 1 (sivo), 3 (RGB paleta), 4 (CMYK paleta) brojeva između 0 i 1. Npr. `\BG{1 0 0}` predstavlja crvenu pozadinu	`\BG{}` (transparentno)
`\CA`	Koristi se kod elemenata koji spadaju u kategoriju gumba. Kod push button elementa predstavlja naziv koji se upisuje u sam element.	`\CA{}`
`\Q`	Poravnanje kod tekstualnih polja. Vrijednosti mogu biti: 0 (lijevo poravnanje), 1 (centriranje teksta), 2 (desno poravnanje)	`\Q{}`(lijevo poravnanje)

\textFont \textFont	Font koji se koristi za prikaz teksta	\textFont{Helv}
\textSize	Veličina teksta	\textSize{9}
\textColor	Boja teksta. Mogu se predavati broj i oznaka palete (g za greyscale i rg za RGB). Npr. \textColor {1 0 0 rg} označava crvenu boju teksta. Moguće je upisati i samo brojeve bez predavanja oznake palete boja (\textColor{1 0 0})	\textColor {0 g}
\MaxLen	Maksimalna duljina teksta koje stane u tekstualno polje.	\MaxLen{}
\symbolchoice	Definira simbol koji se koristi check box ili radio button elemenata. Moguće vrijednosti su check, circle, cross, diamond, square,star.	\symbolchoice{check}

JavaScript

Acrobat JavaScript je skriptni jezik podržan na raznim platformama. Od Acrobat verzije 5.0 Acrobat JavaScript je baziran na JavaScript verziji 1.5 (po ISO-16262) te omogućava nadogradnje za manipulaciju Acrobat formi, stranica, dokumenata te čak i preglednika Acrobat dokumenata.
AcroTEX eDucation Bundle sadrži opsežnu podršku za JavaScript.
Sinkatksa za pisanje JavaScript akcija je sljedeća:
`\pushButton[\A{/S/JavaScript/JS(<JavaScript Code>)}]{jsEx}{22bp}{11bp}`.
Pošto je navedena funkcionalnost podrške za korištenje JavaScripta dosta raširena Acrotex definira `\JS` naredbu pa se gornji primjer može lakše napisati kao:
`\pushButton[\A{\JS{<JavaScript Code>}}]{jsEx}{22bp}{11bp}`.

Akcije se mogu dodavati u elemente forme kroz prvi opcijski parameter. JavaScript akcije se mogu dodavati koristeći `\A` i `\JS` naredbe. Jednostavni JavaScript kôd se može ubaciti koristeći `\JS` naredbu direktno u sam element.

```
\pushButton[\CA{Alert}\A{\JS{%
app.alert("Javascript alert je uspješno izvršen");
}}]{jsSum}{22bp}{11bp}
```

Za ubacivanje kompleksnijeg JavaScript kôda koristi se paket `insdljs`.

1.2 Paket *insDLJS*

Paket definira novu okolinu (environment), `insDLJS`, koji se koristi za ubacivanje Acrobat JavaScripta u pdf datoteke koje su kreirane koristeći Latex.
Paket `insDLJS` ima sljedeće driver opcije:

- `dvipsone`,
- `dvips`,
- `pdftex`,
- `dvipdfm`,
- `dvipdfmx`,
- `xetex`,
- `textures`.

Primjer korištenja paketa za `pdftex driver`:
`\usepackage[pdftex]{insdljs}`.

Lokalne i globalne varijable i funkcije

JavaScript funkcije i varijable se mogu definirati na nivou dokumenta (document level). Kada se PDF dokument otvori prolazi se po funkcijama na nivou dokumenta te iste postaju dostupne kroz čitav dokument što znači da bilo koji Acrobat element može pozvati funkcije koje su definirane na nivou dokumenta.
Varijable koje se definiraju unutar JavaScript funkcija su dostupne samo lokalno (samo unutar funkcije u kojojsu definirane). No, osim funkcija, varijable se isto mogu definirati na nivou dokumenta te su varijable dostupne svima Acrobat elementima te drugim JavaScript funkcijama.
Za primjer uzmimo da se navedeni dio kôda definirao na nivou dokumenta:

```
var globalnaVarijabla = 17;
function Primjer()
{
var lokalnaVarijabla = 3;
app.alert("Pozvana je funkcija Primjer");
}
```

I funkcija `Primjer()` i `globalnaVarijabla` su dostupne kroz čitav dokument. Bilo koji Acrobat element koji izvodi neki JavaScript kôd može pristupiti navedenoj funkciji i varijabli. Za razliku od varijable `globalnaVarijabla`, `lokalnaVarijable` se nalazi unutar funkcije i njoj se ne može direktno pristupiti.

insDLJS okolina

Sintaksna za korištenje `insDLJS` okoline je sljedeća:

```
\begin{insDLJS}[<js_var>]{<base_name>}{<script_name>}
      <Javascript funkcije ili izloženi kôd>
...
...
\end{insDLJS}
```

Okolina uzima `<base_name>` i kreira datoteku `<base_name>.djs`. Datoteka sadrži JavaScript koji je upisan te još dodatan kôd koji okolina može interpretirati. Navedena

datoteka se onda učitava `\AtBeginDocument`.
`insDLJS` je doslovna okolina pri čemu znakovi \ i % zadržavaju postojeće Latex značenje. Okolina se definira kroz 3 parametra:

- <js_var> - opcijski parametar. Ukoliko se radi o `dvipsone` i `dvips` opcijama, a parametar nije naveden, isti se automatski generira. U svim ostalim opcijama (različitim od `dvipsone` i `dvips`) isti se ignorira. Njegov naziv treba biti jednak nazivu jedne od JavaScript funkcija ili JavaScript varijabli definiranih u okolini, a služi za provjeru da li je DLJS već učitan od strane Acrobata.
- <base_name> - vrijednost parametra je limitirana na maksimalno 8 znakova te može sadržavati samo alfanumeričke znakove bez razmaka. Koristi se za naziv pomoćnih datoteka koje se kreiraju te za nazive makora koje koristi okolina.
- <script_name> - naziv JavaScript-a koji se uključuje u dokument. Taj naziv se pojavljuje u dijalogu za JavaScript unutar Acrobat produkta.

Komentari unutar *insDLJS* okoline

Unutar `insDLJS` okoline postoje dva načina na koji se kôd može komentirati:

- TEX komentar - % znak,
- JavaScript komentar,
  - `//` - za komentiranje jedne linije,
  - `/* ... */` - za komentiranje više linija kôda.

Ukoliko se unutar JavaScript funkcije želi koristiti % treba se ispred znaka % staviti znak \ (npr. \%).

Prekidni (escape) znakovi

I JavaScript i TEX koriste znak \ kao prekidni (escape) znak. Paket `insDLJS` pokušava napraviti tranziciju iz TEX-a u JavaScript što je lakše moguće. U sljedećoj tablici se nalazi reprezentacija korisnih znakova.

Sekvenca	Znak koji se reprezentira sa sekvencom
\t	Hotizontalni tab (\u0009)
\n	Novi redak (\u000A)
\r	Oznaka kraja retka (\u000D)
\"	Dvostruki navodnik (\u0022)
\'	Jednostruki navodnik (\u0027)
\\	Obrnuta kosa crta (\u005C)

Unutar JavaScript kôda za te specijalne znakove se trebaju koristiti dvostuki prekidni znakovi.
Npr.

```
\pushButton[\CA{Push Me}\A{\JS{%
app.alert(Primjer \\"teksta unutar navodnika\\" . );}}]
{primjerEsc}{}{10bp};
```

Traženje pogrešaka (debugging)

Pisanje JavaScript kôda te traženje pogrešaka može biti dosta kompleksno ukoliko se ne posjeduje Acrobat softver. U besplatnom programu Acrobat Reader nema alata za debagiranje JavaScript kôda unutar dokumenta te se traženje pogrešaka svodi na ispis poruka (alert) za svaki korak algoritma koji se implementira što uvelike otežava rad te zahtjeva puno vremena da bi se pronašle i ispravile eventualne pogreške.
Ukoliko se posjeduje Acrobat softver implementacija i traženje pogrešaka je uvelike olakšano zato što Acrobat ima GUI editor koji odmah provjerava i sintaksne pogreške.

1.3 O eksponencijalnim funkcijama

U dokumentu [3] dani su sljedeći ishodi učenja vezani uz eksponencijalnu funkciju:
Učenici će:

- računati s potencijama,
- opisati i izvesti jednostavne ovisnosti (veze) dviju veličina formulama, tablicama, grafovima i riječima; prevesti s jednoga od navedena četiri oblika na drugi te čitati, uspoređivati i tumačiti ovisnosti (veze)
- prepoznati, odrediti i protumačiti karakteristične elemente i svojstva jednostavnih funkcija, analizirati linearne, kvadratne, eksponencijalne, logaritamske i trigonometrijske funkcije te rabiti njihova svojstva
- računski, grafički i uz pomoć računala, u skupu realnih brojeva riješiti linearne, kvadratne, eksponencijalne i logaritamske jednadžbe i nejednadžbe i sustave jednadžba
- primijeniti funkcije i njihove grafove te jednadžbe i nejednadžbe u rješavanju matematičkih problema i problema u ostalim odgojno-obrazovnim područjima i svakodnevnomu životu.

U srednjim školama se ovi ishodi ostvaruju obradom odgovarajućih matematičkih sadržaja u bar dvije školske godine. Prvo se u 1. razredu učenici susreću s pojmom potencija. Obrađuju se one potencije kojima je eksponent cijeli ili racionalni broj. U 2. razredu srednje škole to znanje produbljuju uvodeći pojam eksponencijalne funkcije.
Moramo napomenuti da njihov susret s potencijama u 1. razredu srednje škole nije prvi. Naime, već u matematici osmog razreda obrađuju se potencije s bazom 10, a u fizici sedmog i osmog razreda je uobičajena praksa računati s brojevima zapisanima u znanstvenom zapisu.

Pojam potencije uvodi se postupno: od potencije s prirodnim eksponentom pa sve do potencije s realnim eksponentom. Tako se potencija s prirodnim eksponentom definira ovako:
Za $a > 0$, $a \in \mathbb{R}$ i $n \in \mathbb{N}$ imamo
$a^2 = a \cdot a$, $a^3 = a \cdot a \cdot a = a^2 \cdot a$, ... $a^n = \underbrace{a \cdot a \cdot ... \cdot a}_{\text{n puta}} = a^{n-1} \cdot a$.
Broj a nazivamo baza, a n eksponent potencije a^n.
Osnovna svojstva ovako definiranih potencija slijede direktno iz definicije:

1. $a^x . ^y = a^{x+y}$,

2. $(a^x)^y = a^{x \cdot y}$,

3. $(a \cdot b)^x = a^x \cdot b^x$.

Nakon što su učenici usvojili potencije kojima je eksponent prirodan broj, uvode se potencije čiji je eksponent negativan cijeli broj:

$$a^{-n} = \frac{1}{a^n}, a > 0 \text{ i } n \in \mathbb{N}.$$

Uz dogovor da je $a^0 = 1$ u potpunosti su definirane potencije s cjelobrojnim eksponenetom. Potenciranje pozitivnog broja a recipročnim brojem prirodnog broja n definira se na sljedeći način:

$$a^{\frac{1}{n}} = \sqrt[n]{a}.$$

Nakon toga zanima nas je što se događa kada je baza a pozitivan broj, a eksponent bilo koji racionalan broj. Ako je $m \in \mathbb{Z}$, $n \in \mathbb{N}$, $a^{\frac{m}{n}}$ definira se ovako:

$$a^{\frac{m}{n}} = \sqrt[n]{a^m}.$$

Za sve ove potencije vrijede prethodno navedena svojstva i na taj način je definirana potencija a^n za sve pozitivne brojeve a te racionalne brojeve x. Ukoliko je eksponent x iracionalni broj tada se potencija a^x uvodi pomoću donjih i gornjih aproksimacija. Obično se taj postupak ilustrira na nekoliko konkretnih primjera. Na primjer, neka je eksponent $x = \sqrt{2}$, a baza $a = 10$. Broj $\sqrt{2}$ je iracionalan, ali niz njegovih donjih i gornjih aproksimacija je ovakav:

$$\begin{aligned} 1 &< \sqrt{2} < 2 \\ 1.4 &< \sqrt{2} < 1.5 \\ 1.41 &< \sqrt{2} < 1.42 \\ 1.414 &< \sqrt{2} < 1.415 \end{aligned}$$

Tada broju $10^{\sqrt{2}}$ pristupamo kao broju koji je ugniježđen između donje i gornje aproksimacije, tj.

$$\begin{aligned} 10^1 &< 10^{\sqrt{2}} < 10^2 \\ 10^{1.4} &< 10^{\sqrt{2}} < 10^{1.5} \\ 10^{1.41} &< 10^{\sqrt{2}} < 10^{1.42} \\ 10^{1.414} &< 10^{\sqrt{2}} < 10^{1.415} \end{aligned}$$

Lijeve i desne strane gornjih vrijednosti su potencije s racionalnim eksponentom koje su već definirane. Jasno je da se pri ovakvom pristupu implicitno koristi rast (za bazu veću od 1) funkcije koju ćemo kasnije definirati kao eksponencijalnu funkciju, te koristimo i vrlo netrivijalnu činjenicu da je broj $10^{\sqrt{2}}$ definiran ovako jedinstven.
Konačno, postupak definiranja potencija završava definiranjem novog pojma, a to je eksponencijalna funkcija i to se obično radi u 2. razredu srednje škole.
Ono što je dalje u fokusu pažnje u 2. razredu srednje škole jeste crtanje grafova eksponencijalnih funkcija.
Pored osnovnih svojstava koja su već proučavana u 1. razredu kao svojstva potencija, sad se intenzivno proučava monotonost funkcije $x \mapsto a^x$, što će naći svoju primjenu pri rješavanju eksponencijalnih nejednadžbi te pri definiranju logaritamske funkcije koja se u gimnazijama obično definira kao inverzna funkcija eksponencijalne.

Graf i svojstva eksponencijalne funkcije

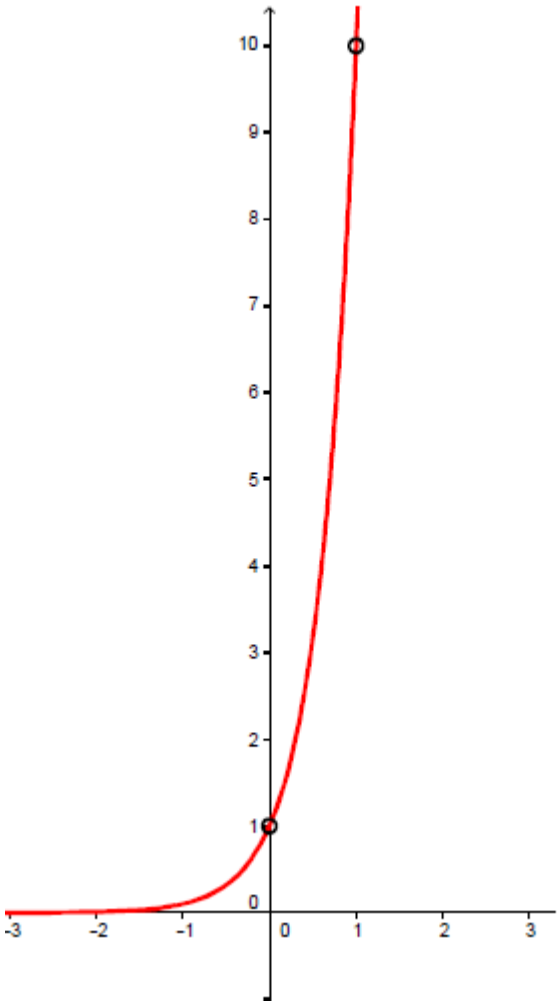
Pri crtanju grafa eksponencijalne funkcije krećemo od konkretnih funkcija: $f(x) = 10^x$, $f(x) = 2^x$, $f(x) = 3^x$ i njihovih tablica vrijednosti, te izvodimo zaključak o izgledu opće eksponencijalne funkcije $f(x) = a^x$, $a > 1$. Potom se razmatra slučaj kad je $0 < a < 1$ uz stalno isticanje kad funkcija raste odnosno pada.

Graf funkcije $x \mapsto 10^x$

Prije skiciranja grafa funkcije $f(x) = 10^x$ napravit ćemo tablicu u koju ćemo zapisati njezine vrijednosti za nekoliko odabranih vrijednosti x.

x	10^x
-3	$10^{-3} = 0.001$
-2	$10^{-2} = 0.01$
-1	$10^{-1} = 0.1$
0	$10^0 = 1$
0.5	$10^{0.5} = \sqrt{10} = 3.16$
1	$10^1 = 10$
1.5	$10^{1.5} = \sqrt{1000} = 31.6$
2	$10^2 = 100$

Brojevi $\sqrt{10}$ i $\sqrt{1000}$ su iracionalni pa je vrijednost u točkama 0.5 i 1.5 izračunata približno.

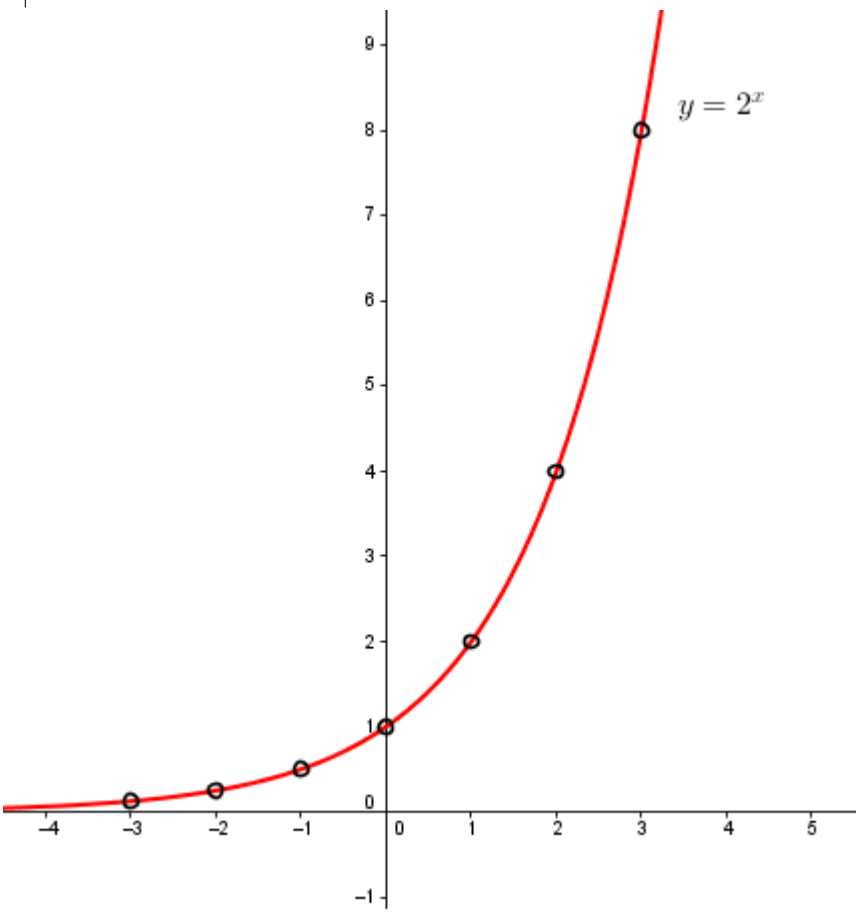


S grafa je očito da eksponencijalna funkcija raste vrlo brzo za pozitivne brojeve x. Ako crtamo u mjerilu $1:1$ za $x = 10$ cm koordinata y je 10^{10} cm= 10^5 km. Isto tako, funkcija brzo pada za negativne argumente x. Negativni dio osi x je asimptota grafa eksponencijalne funkcije.

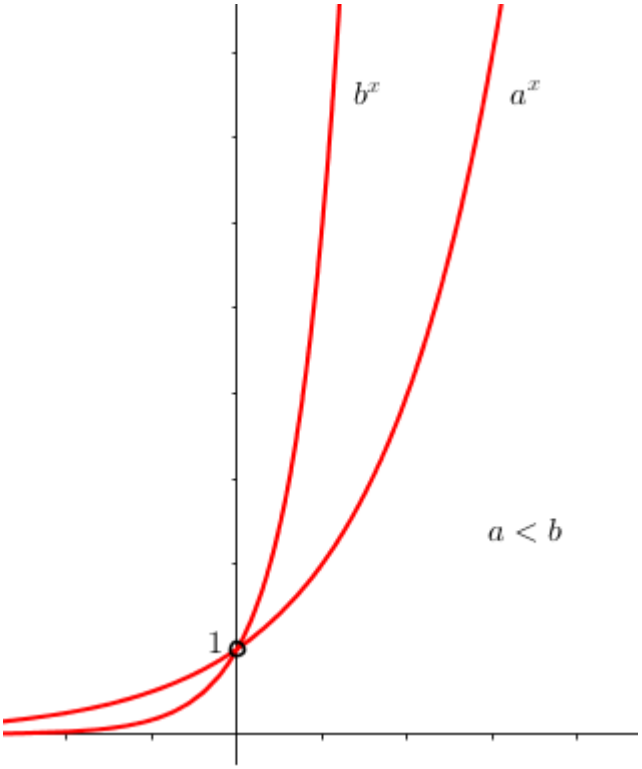
Graf eksponencijalne funkcije $x \mapsto a^x$

Osim za $a = 10$, graf funkcije a^x se može nacrtati i za druge vrijednosti baze a.
Opet prvo radimo tablicu u koju ćemo zapisati odabrane točke te vrijednosti funkcije u tim točkama.

x	2^x
-3	$\frac{1}{8}$
-2	$\frac{1}{4}$
-1	$\frac{1}{2}$
0	1
1	2
2	4
3	8



Kao što se može vidjeti na sljedećem grafu, ta funkcija ima sličan graf kao funkcija $x \mapsto 10^x$. Zato jer je $2^x < 10^x$ za $x > 0$, graf funkcije 2^x sporije raste. Za negativne brojeve x vrijedi suprotna nejednakost.



Graf eksponencijalne funkcije s bazom $0 < a < 1$

Graf eksponencijalnih funkcija s bazom $0 < a < 1$ se može izvesti iz grafa eksponencijalnih funkcija s bazom većom od 1, a to je nešto što su učenici već dosad savladali.
U istom koordinatnom sustavu prikazat ćemo grafove funkcija $f(x) = 2^x$ i $g(x) = 2^{-x}$.

x	2^x	2^{-x}
-3	$\frac{1}{8}$	8
-2	$\frac{1}{4}$	4
-1	$\frac{1}{2}$	2
0	1	1
1	2	$\frac{1}{2}$
2	4	$\frac{1}{4}$
3	8	$\frac{1}{8}$

Grafovi ovih funkcija su simetrični s obzirom na y os jer one poprimaju iste vrijednosti za brojeve suprotnih predznaka.

Injektivnost eksponencijalne funkcije

Svojstvo monotonosti eksponencijalne funkcije navodi nas na jedan jako važan zaključak, a to je:

$$\text{Ako je } a^{x_1} = a^{x_2}, \text{ onda vrijedi } x_1 = x_2.$$

1.4 Primjeri korištenja paketa *eforms* i *insDLJS*

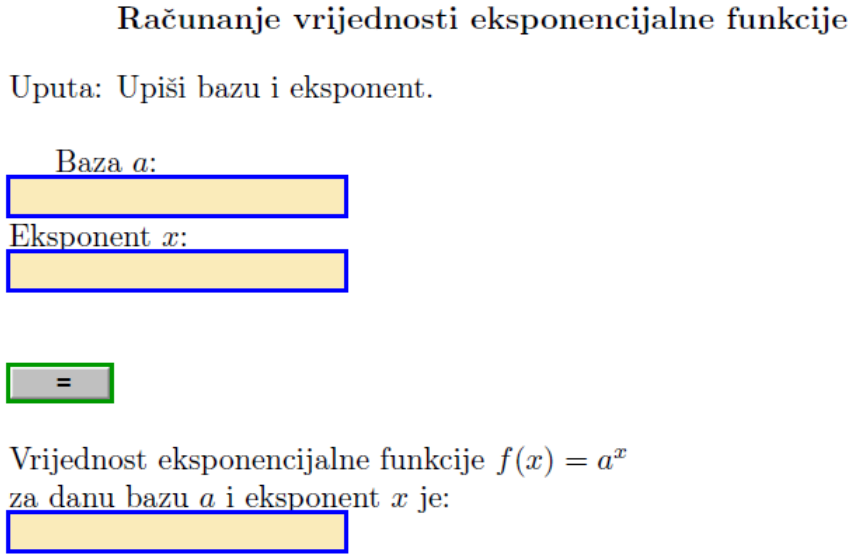
Eksponencijalne funkcije

U nastavku se nalazi primjer pdf kalkulatora koji koristi pakete `eforms` i `insDLJS` za izračun vrijednosti eksponencijalnih funkcija.
Koraci:

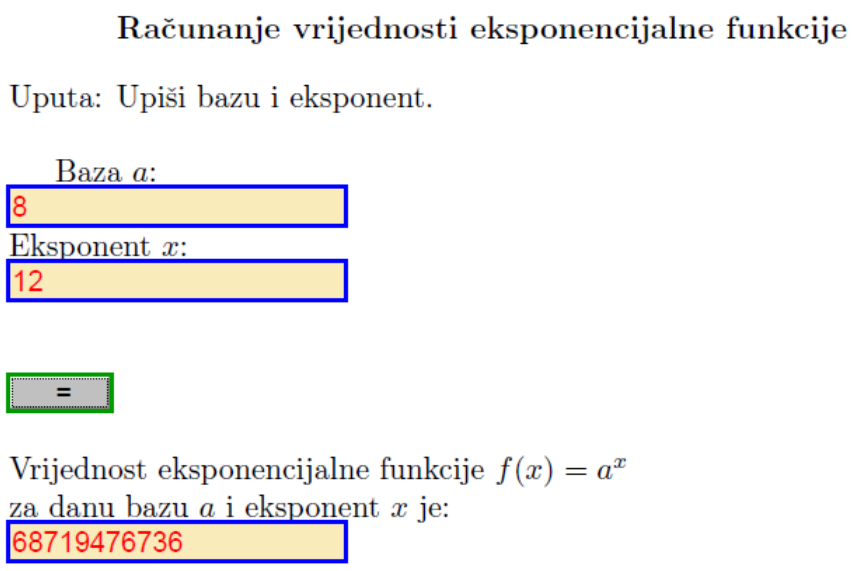
1. Korisnik unese bazu u Text Field
2. Korisnik unese eskponent u Text Field
3. Korisnik klikne na gumb koji na sebi ima oznaku =
4. Sustav izračuna rezultat koristeći JavaScript funkciju

```
\documentclass{article}
\usepackage[designi]{web}
\usepackage[pdftex]{eforms}
\usepackage[pdftex]{insdljs}
\usepackage[cp1250]{inputenc}

\begin{insDLJS}{eksdljs}{skriptaZaRacunanje}
function racunaj() {
var baza = this.getField("baza").value;
var eksponent = this.getField("eksponent").value;
this.getField("rezultat").value = Math.pow(baza, eksponent);
}
\end{insDLJS}

\begin{document}
\begin{center}
\textbf{ Računanje vrijednosti eksponencijalne funkcije }
\end{center}
Uputa: Upiši bazu i eksponent.\\

     Baza $a$:

    \textField [\BC{0 0 1}\BG{0.98 0.92 0.73}\textColor{1 0 0}
]{baza}{1.5in}{12bp} \\
     Eksponent $x$:

    \textField [\BC{0 0 1}\BG{0.98 0.92 0.73}
\textColor{1 0 0}
]{eksponent}{1.5in}{12bp} \\\\

    \pushButton[\BC{0 .6 0}\CA{ = }
\A{\JS{racunaj();}}]{racunajBtn}{33bp}{11bp}
    \\\\
    Vrijednost eksponencijalne funkcije $f(x)=a^x$
    \\za danu bazu $a$ i eksponent $x$ je: \\
  \textField [\BC{0 0 1}\BG{0.98 0.92 0.73}
\textColor{1 0 0}
]{rezultat}{1.5in}{12bp}\\
\end{document}
```

Primjer izgleda pdf kalkulatora za ovaj slučaj:



Kad unesemo neke brojke, izgleda ovako:



Naravno, postoji ograničenje:

Računanje vrijednosti eksponencijalne funkcije

Uputa: Upiši bazu i eksponent.

Baza a:
80000
Eksponent x:
20

=

Vrijednost eksponencijalne funkcije $f(x) = a^x$
za danu bazu a i eksponent x je:
1.152921504606847e+98

Ali, u jednom trenutku više ne zna računati i kod jako velikih brojeva za odgovor nudi: Infinity.

Računanje vrijednosti eksponencijalne funkcije

Uputa: Upiši bazu i eksponent.

Baza a:
8000000
Eksponent x:
200

=

Vrijednost eksponencijalne funkcije $f(x) = a^x$
za danu bazu a i eksponent x je:
Infinity

Zbrajanje i oduzimanje matrica

Drugi primjer kalkulatora je kalkulator za zbrajanje i oduzimanje matrica. Prvi problem koji se javlja je način unosa; kao prvo, korisnik može bit zbunjen kako da unosi matricu u polje a kao drugo, ne može JavaScript prihvatiti različite načine unosa. Zato je prije definiranja funkcije, potrebno izabrati način i te upute smo stavili na prvu stranu kalkulatora:

Upute:
Podaci se moraju unositi u sljedećem formatu:
$[x;y;z;w][w;t;r;p][a;b;c;d]$ – ovo je primjer matrice $3x4$.
Pri čemu su pravila sljedeća:

- otvorena uglata zagrada označava početak reda u matrici;
- zatvorena uglata zagrada označava kraj reda u matrici;
- između brojeva ide zarez i nigdje ne smije biti razmaka;
- unose se samo cijeli brojevi.

```
\documentclass{scrartcl}
\usepackage{amsmath}
\usepackage[designi]{web} % dvips, pdftex, dvipsone, dvipdfm
\usepackage{exerquiz}
\usepackage[ImplMulti,indefIntegral]{dljslib}
\usepackage[max=3]{renditions}
\usepackage[pdftex]{insdljs}
\usepackage{mathptmx}
\usepackage[utf8]{inputenc}
\usepackage[croatian]{babel}
\thispagestyle{empty}
```

```
\begin{insDLJS}[test]{test}{JavaScript}
 var delimiterPocetkaRetka = "[";
 var delimiterKrajaRetka = "]";
 var delimiterElemenata = ",";
```

```
 var noviRedak = "\r";


function parseMatrica(fieldname) {

 var matricaVrijednost =
 this.getField(fieldname).value;

    // micanje svih novih redova
 var regexpNoviRedak = new RegExp(noviRedak, 'g');
 matricaVrijednost =
 matricaVrijednost.replace(regexpNoviRedak, '');

 var array =
 matricaVrijednost.split(delimiterKrajaRetka);

 var arrayLength = array.length;
 var redakMatriceKaoString = [];
 var redakStrBrojac = 0;
 for (var i = 0; i < arrayLength; i++) {
     if (array[i].length > 0) {
      redakMatriceKaoString[redakStrBrojac]
      = array[i].substring(1, array[i].length);
      redakStrBrojac++;
     }
 }

 var matrica = [];
 for (var j = 0; j < redakMatriceKaoString.length; j++)
  {
     if (redakMatriceKaoString[j].length > 0) {
      matrica[j] =
      redakMatriceKaoString[j].split(delimiterElemenata);
     }
 }
 return matrica;
};
```

```
function zbrojiMatrice(poljeZaRezultat) {
var matrica1 = parseMatrica("matrica1");
var matrica2 = parseMatrica("matrica2");

if (provjeriDimenzijeMatrica
(matrica1, matrica2) == false)
return false;

var zbrojMatrica = [];
var redakMatrica1 = [];
var redakMatrica2 = [];
for (var i = 0; i < matrica1.length; i++)
{
redakMatrica1 = matrica1[i];
redakMatrica2 = matrica2[i];

var redakZbrojenaMatrica = [];
for (var j = 0; j < redakMatrica1.length; j++)
 {
var x = parseInt(redakMatrica1[j]);
var y = parseInt(redakMatrica2[j]);
redakZbrojenaMatrica[j] = x + y;
}

zbrojMatrica[i] = redakZbrojenaMatrica;
}

var rezultatKaoString =
 formatirajMatricuZaIspis(zbrojMatrica);
this.getField(poljeZaRezultat).value =
rezultatKaoString;
};

function oduzmiMatrice(poljeZaRezultat) {
var matrica1 = parseMatrica("matrica3");
var matrica2 = parseMatrica("matrica4");

if (provjeriDimenzijeMatrica
(matrica1, matrica2) == false)
```

```
return false;

var razlikaMatrica = [];
var redakMatrica1 = [];
var redakMatrica2 = [];
for (var i = 0; i < matrica1.length; i++) {
redakMatrica1 = matrica1[i];
redakMatrica2 = matrica2[i];

var redakRazlikaMatrica = [];
for (var j = 0; j < redakMatrica1.length; j++)
{
var x = parseInt(redakMatrica1[j]);
var y = parseInt(redakMatrica2[j]);
redakRazlikaMatrica[j] = x - y;
}

razlikaMatrica[i] = redakRazlikaMatrica;
}

var rezultatKaoString =
formatirajMatricuZaIspis(razlikaMatrica);
this.getField(poljeZaRezultat).value
= rezultatKaoString;
};

function provjeriDimenzijeMatrica
(matrica1, matrica2) {
if (matrica1.length != matrica2.length){
app.alert("Matrice nisu istih dimenzija");
return false;
}

var redakMatrica1 = [];
var redakMatrica2 = [];
for (var i = 0; i < matrica1.length; i++) {
redakMatrica1 = matrica1[i];
redakMatrica2 = matrica2[i];
```

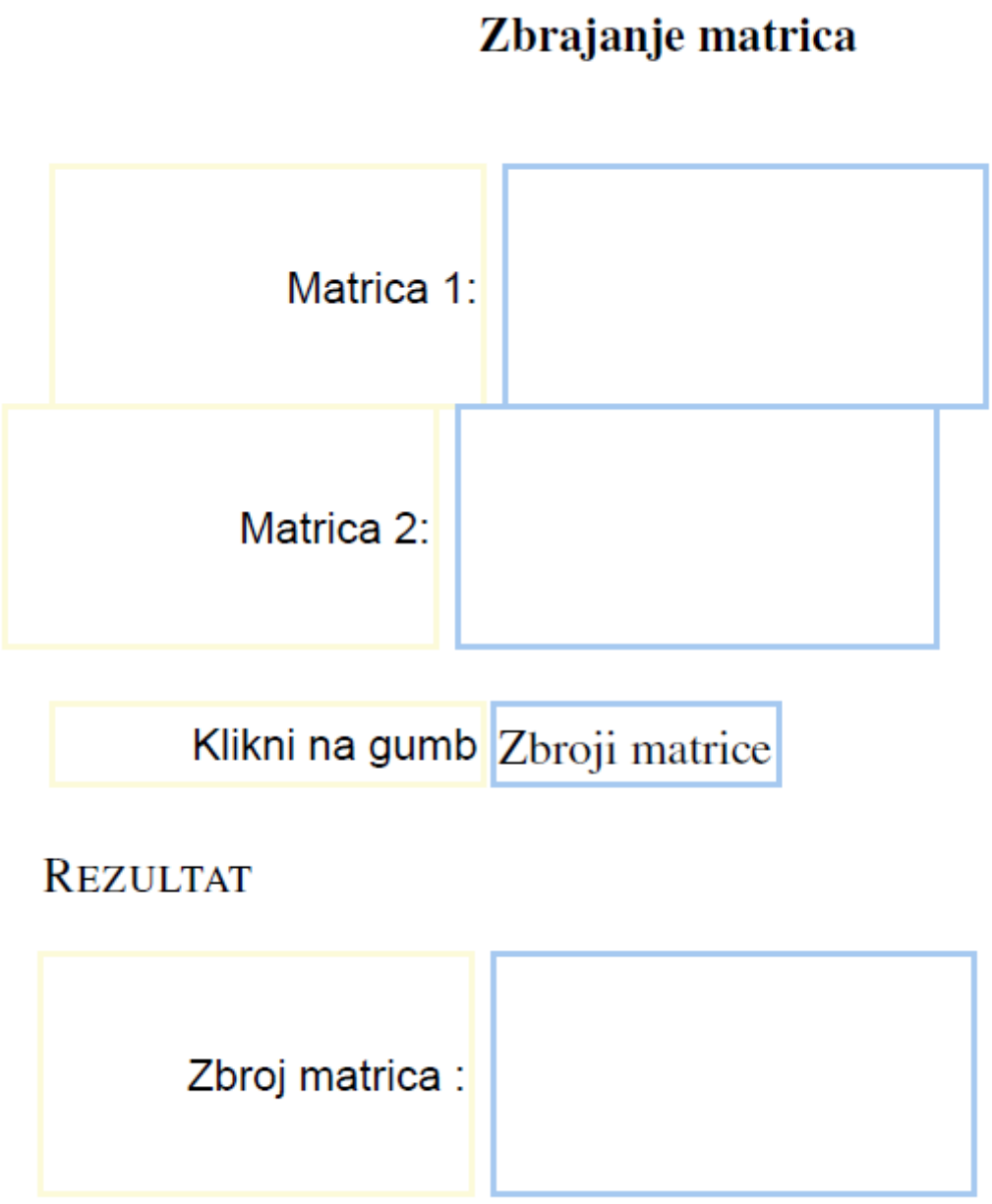
```
if (redakMatrica1.length
!= redakMatrica2.length) {
app.alert
("Matrice nisu istih dimenzija");
return false;
}
}
return true;
};

function formatirajMatricuZaIspis(matrica) {
var matricaZaIspis = "";

for (var i = 0; i < matrica.length; i++) {
matricaZaIspis += delimiterPocetkaRetka;
var redak = matrica[i];

for (j = 0; j < redak.length; j++) {
matricaZaIspis += redak[j];
if (j < redak.length - 1)
matricaZaIspis
+= delimiterElemenata
}
matricaZaIspis =
matricaZaIspis + delimiterKrajaRetka + noviRedak;
}

return matricaZaIspis;
}
\end{insDLJS}

%% Short hand commands
\newcommand{\textforlabel}[2]{%
\TextField[name={#1}, value={#2}, width=9em,align=2,%
bordercolor={0.990 .980 .85},%
readonly=true]{}%
}

\newcommand{\heading}[1]{\textsc{#1}}
```

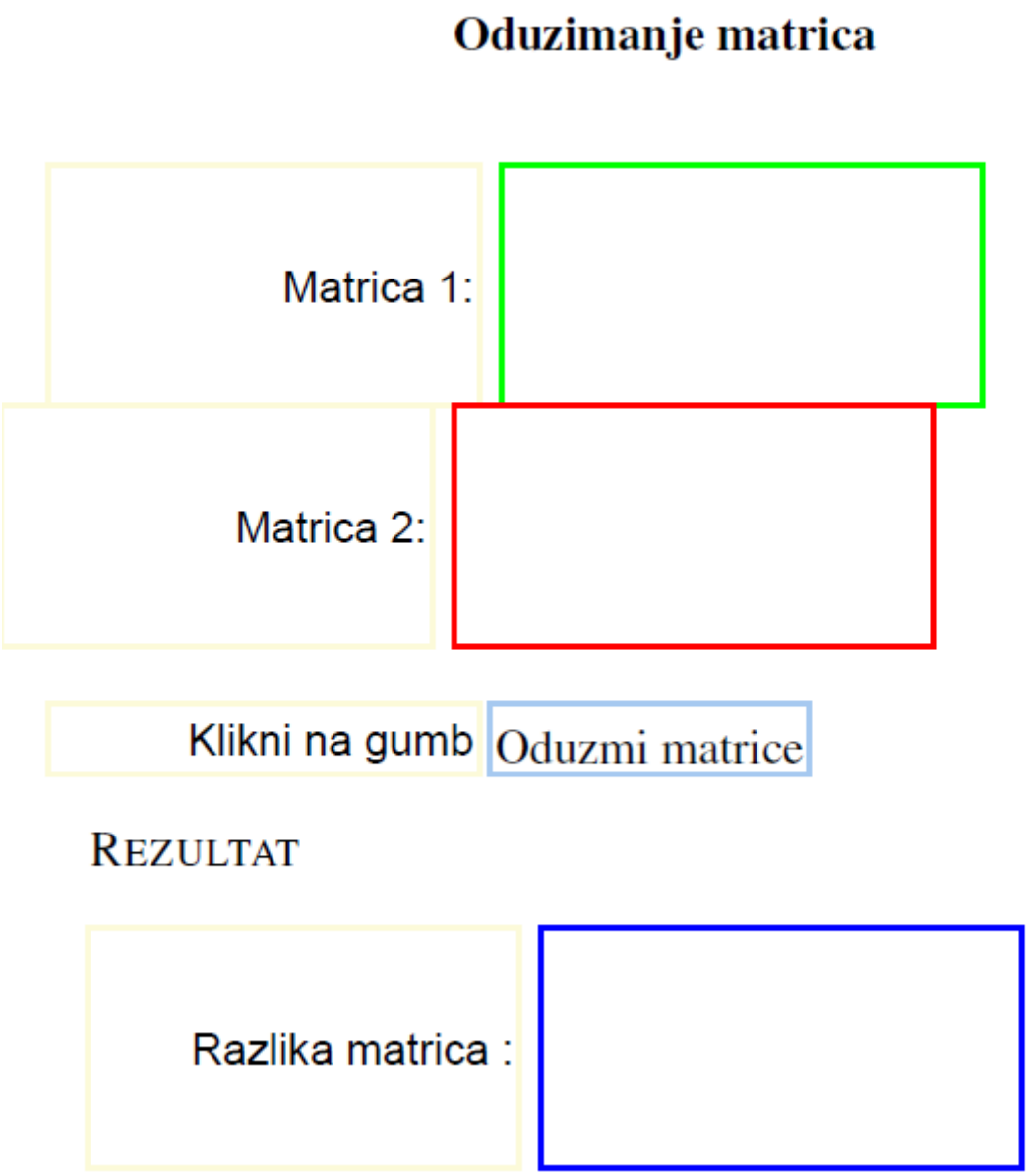
```
\begin{document}
Upute:\\
Podaci se moraju unositi u sljedećem formatu:\\
$[x;y;z;w][w;t;r;p][a;b;c;d]$ {
ovo je primjer matrice $3x4$.\\
Pri čemu su pravila sljedeća:
\begin{itemize}
\item otvorena uglata zagrada označava
početak reda u matrici;
\item zatvorena uglata zagrada označava
kraj reda u matrici;
\item između brojeva ide zarez i
nigdje ne smije biti razmaka;
\item unose se samo cijeli brojevi.
\end{itemize}

\newpage

\begin{center}
\textbf{ Zbrajanje matrica }
\end{center}

\begin{Form}

\heading{}

\textforlabel{01}{Matrica 1:}
\TextField[name=matrica1,width=10em,
bordercolor={0.650 .790 .94}, multiline = true]{}%
\\
\textforlabel{101}{Matrica 2:}
\TextField[name=matrica2,width=10em,
bordercolor={0.650 .790 .94}, multiline = true]{}%
\\

\textforlabel{l02}{Klikni na gumb }
\PushButton[name=start,onclick={zbrojiMatrice
```

```
('zbrojMatrica');},bordercolor={0.650 .790 .94}%
]{Zbroji matrice}\\

\newpage

\heading{Rezultat}\\

%% RESULTS
%% Diameter
\textforlabel{name=l04}{%
Zbroj matrica :} \TextField
[name=zbrojMatrica,width=10em,
bordercolor={0.650 .790 .94},%
readonly=true, multiline = true]{}

\end{Form}

\newpage

\begin{center}
\textbf{ Oduzimanje matrica }
\end{center}
\begin{Form}

\heading{}

\textforlabel{02}{Matrica 1:}
\TextField[name=matrica3,width=10em,
bordercolor={0 1 .0}, multiline = true]{}%
\\
\textforlabel{201}{Matrica 2:}
\TextField[name=matrica4,width=10em,
bordercolor={1 0 0}, multiline = true]{}%
\\

\textforlabel{202}{Klikni na gumb }
\PushButton[name=oduzmi,onclick=
{oduzmiMatrice('razlikaMatrica');},
bordercolor={0.650 .790 .94}%
```

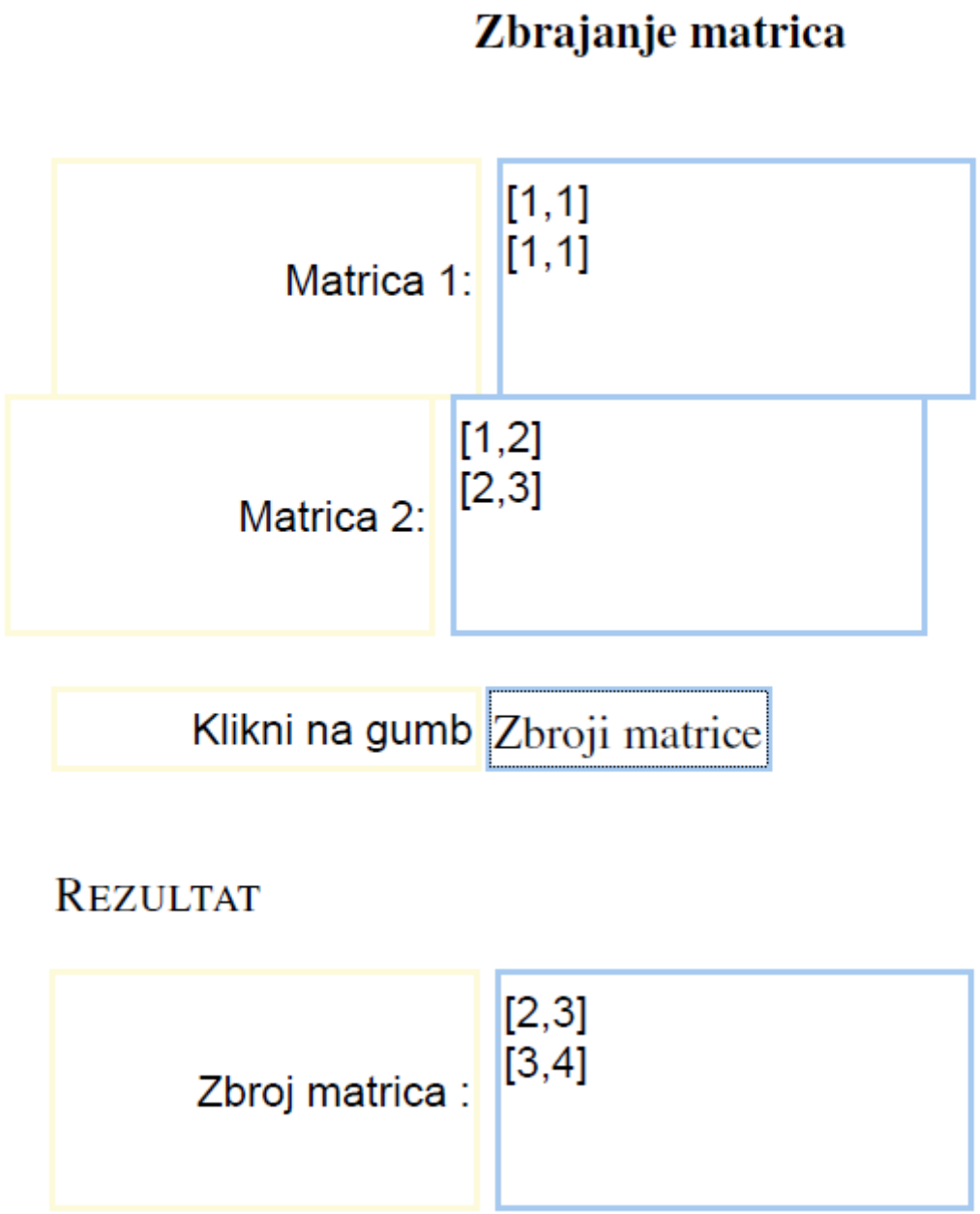
```
]{Oduzmi matrice}\\

\newpage

\heading{Rezultat}\\

%% RESULTS
%% Diameter
\textforlabel{name=204}{%
Razlika matrica :} \TextField
[name=razlikaMatrica,width=10em,
bordercolor={0 0 1},%
readonly=true, multiline = true]{}

\end{Form}



\end{document}
```

S prethodnim kôdom dobijemo ovakav kalkulator:

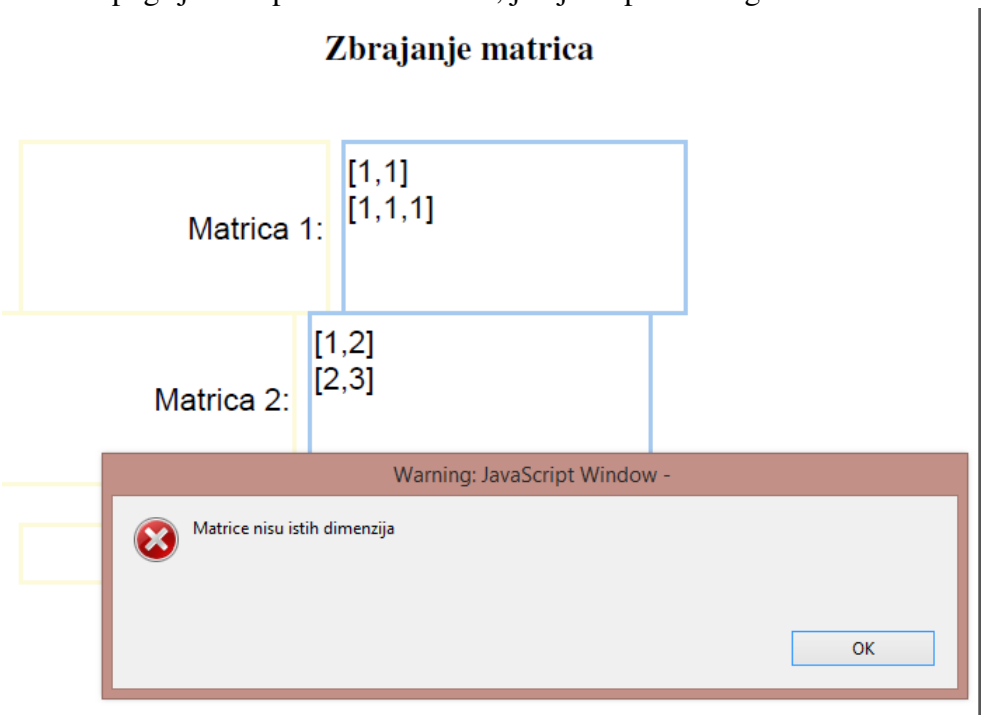




Polja za unos su namjerno u različitim bojama da se vidi kako se može manipulirati i određivati izgled.
Nakon unosa matrice i klika na gumb ”Zbroji matrice”, imamo ovakav izračun:


Zbrajanje matrica
Matrica 1:
[1,1]
[1,1]
Matrica 2:
[1,2]
[2,3]
Klikni na gumb
Zbroji matrice
REZULTAT
Zbroj matrica :
[2,3]
[3,4]

Ako pogriješimo pri unosu matrica, javlja se poruka o greški.



Poglavlje 2

Paket eDucation Bundle

2.1 Uvod

The Acrotex eDucation Bundle je skup LaTex makro datoteka zajedno s različitim datotekama podrške i primjera. Ovaj paket je namijenjen izradi e-publikacija u obrazovanju koristeći LaTex editor kao aplikaciju u koju se unosi sadržaj, a kao izlazni dokument dobiva se pdf datoteka.
Trenutno, postoje nekoliko komponenti paketa:

1. Paket `web` koji se koristi za izradu zanimljivih, ugodnih oku stranica prikladnih za `www` ili prezentacije.
2. Paket `exerquiz` na jednostavan način kreira interaktivne kvizove i vježbe.
3. Paket `eforms` pruža podršku za PDF dokomente u obliku raznih formi za popunjavanje.
4. Paket `insdljs` dozvoljava automatsko ubacivanje dokumenata na razini JavaScripta. Autor može koristiti `insdljs` za kreiranje kvizova u paketu `exerquiz` .
5. Paket `dljslib` se koristi kao osnovni skup JavaScript funkcija.
6. Paket `eqExam` je samostalan LaTex paket za izradu ispita, kvizova, zadaća. Može se koristiti i za izradu anketa, upitnika, evaluacije učitelja, itd. Paket ima email funkcionalnost koja dolazi sa skriptama koje se nalaze na serveru te se omogućava slanje e-poruke učitelju s podacima koji su upisani u formu.

AcroTex Bundle bi mogao biti koristan nastavnicima koji žele na internet stavljati razne interaktivne sadržaje na internet, a pogotovo matematičke zbog korištenja LaTeX-a koji je prvobitno i zamišljen kao alat za pisanje matematičkih izraza.

Povijest

Paketi `web` i `exerquiz` su napisani kao priprema za dvodnevnu radionicu na temu LaTeX/PDF koju je na Redwoods sveučilištu u Eureki, Kalifornija održao D. P. Story. Termini radionice su bili od 30. travnja do 1. svibnja 1999. godine.
Za potrebe radionice morao je uzeti mnogo osnovnih makroa koje je radio u osnovnom TeX-u i prebaciti ih u LaTeX.
Značajni dodaci `exerquiz-u` su napravljeni odmah za 20. godišnju konferenciju TeXUser's Group (tug) u kolovozu 1999. godine u Vancouveru.
Paket `insdljs` je napisan za 22. godišnju konferenciju Tex User's Group (tug) koja je održana u kolovozu 2011. na Sveučilištu Delaware u Newarku, Delaware.

2.2 Paket *exerquiz*

Općenito

Paket `exerquiz` daje okolinu za izradu sljedećih elemenata u pdf dokumentu:

- `exercize` okolina: makroi za kreiranje on-line vježbi.
- `shortquiz` okolina: makroi za izradu interaktivnih kvizova s odmah dostupnom povratnom informacijom.
- `shortquiz` s rješenjima: makroi za izradu interaktivnih kvizova s odmah dostupnom povratnom informacijom i s linkom na rješenja kviza.
- `quiz` okolina: makroi za kreiranje kviza koji ocjenjuje JavaScript s mogućnošću da ga JavaScript i ispravlja.

U svakoj od ovih okolina za mogućnost odgovora može se staviti višestruki izbor, matematičko nadopunjavanje i nadopunjavanje tekstom.

Paket *exerquiz* i Acrobat JavaScript

Paket `exerquiz` sada koristi `insdljs` paket za umetanje dokumenta na razini JavaScripta u PDF datoteke. Kvizovi napravljeni pomoću `shortquiz` ili `kviz` okoline se ocjenjuju, označavaju i boduju pomoću umetnutih JavaScript funkcija. Budući da je paket `insdljs` već učitan, autor dokumenta razvija vrlo lako JavaScript koji može biti pozvan standardnom `eexerquiz` naredbom. Mogućnost pisanja JavaScripta u samom LaTex dokumentu daje jedinstvenu sposobnost programiranja u `exerquizu`.

Zahtjevi paketa

Paket `exerquiz` je neovisan o paketu `web`. Međutim, `exerquiz` koristi `hyperref` jednako kao i `web`. Uvijek se mora koristiti najnovija verzija `hyperrefa`. Osim paketa boja, koji se koristi i u `webu`, `exerquiz` također koristi paket `verbatim`. To se koristi za pisanje doslovnih rješenja vježbi i testova za određene pomoćne datoteke.
Rezultati iz kvizova stvorenih u `shortquiz` i `quiz` okolinama vrednuju se koristeći JavaScript. Te JavaScript funkcije su umetnute u konačnu PDF datoteku pomoću paketa `insdljs`. Ovaj paket olakšava autoru da napiše JavaScript.
Paket `exerquiz` koristi značajke pdf-a koje paket `web` ne koristi. Da bi interaktivne značajke radile kako treba, koristi se Adobe Reader 5.0 ili noviji.

2.3 Osnovne opcije paketa

U preambulu LaTeXdokumenta se stavlja
`\usepackage{exerquiz}`
Kod upotrebe `exerquiza` s paketom `web` upisujemo i sljedeće naredbe u preambulu:
`\usepackage[⟨ driver_options⟩,⟨ more_options⟩]{web}`
`\usepackage[⟨ options⟩]{exerquiz}`

Naredbe u preambuli s `dvipsone` ili `dvips` driver opcijama izgledaju ovako:
`\usepackage[⟨ driver_options⟩,⟨ more_options⟩]{hyperref}`
`\usepackage{exerquiz}`

S `driver` opcijama `dvipdfm` koristi se `hyperref` i `exerquiz` i naredbe u preambuli izgledaju ovako:
`\usepackage[⟨ driver_options⟩,⟨ more_options⟩]{hyperref}`
`\usepackage[⟨ driver_options ⟩]{exerquiz}`

Opcija *pdftex*

Paket `exerquiz` ne ovisi o paketu `web`, a s njime možemo kreirati razne vježbe i kvizove. Ukoliko se želi kreirati vlastiti paket za određivanje izgleda stranice, mora se koristiti `hyperref` .
Ako se želi koristiti paket `exerquiz` s `pdftex-om` bez paketa `web`, upotrebljavamo ove naredbe:
`\usepackage[pdftex, ⟨ more_options ⟨]{hyperref}`
`\usepackage[pdftex]{exerquiz}`

Opcija *dvipdfm*

Ako se koristi paket `exerquiz` bez paketa `web`, treba staviti:
`\usepackage[dvipdfm,⟨ more_options⟩]{hyperref}`
`\usepackage[dvipdfm]{exerquiz}`

Napomena 2.3.1. *Ove dvije opcije podržavaju samo* `exercise` *okolinu. Nijedna kviz okolina ne može biti korištena s ove dvije opcije.*

Opcija jezik

Opcija jezik je dostupna u paketu `web`, ali može biti uključena čak i kada se paket `web` ne koristi. Trenutno su podržani nizozemski, francuski, njemački, talijanski, norveški, ruski, španjolski, poljski, finski, češki i katalonski. Na primjer, želimo li uključiti francuski jezik, naredbe izgledaju ovako:
`\usepackage[⟨ driver_option⟩,⟨ more_options⟩]{hyperref}`
`\usepackage[⟨ driver_option⟩, french]exerquiz;`

Opcije *forpaper* i *forcolorpaper*

Kod kreiranja vježbi u `exercize` okolini za ispis na papir možemo koristiti opciju `forpaper`. Ta opcija ukljanja boju iz dokumenta i vraća uobičajeni `\textheight` iz `articlea`- standardne klase dokumenata u LaTeX-u. Opcija `\textwidth` je određena s parametrima `screensize` i `margins` ili s raznim opcijama za uređivanje i zbog toga je npr. prelazak u novi red na istom mjestu i u web verziji i u verziji za printanje.
Koristeći ovu opciju s opcijom `latexlayout` dobivamo standardnu LaTeXopciju `\textwidth`.
Opcija `forpaper` također mijenja naredbu `\newcommand` u `\par\medskip` na kraju svakog rješenja da ne bi trošili više papira za ispis ako nije potrebno.
Za dodatno prerađivanje postoji naredba `ifeqforpaper` koja se koristi kod točno/netočno pitalica.
Jedina razlika kod `forpaper` i `forcolorpaper` opcija je to što druga ne isključuje boje.

Osim što su opcije `forpaper` i `forcolorpaper` dostupne u paketu `web`, potrebne su i u paketu `exerquiz` ako se on koristi bez paketa `web`. Opcija se poziva na uobičajen način:
`\usepackage[⟨ options⟩]{hyperref}` `%` ili `pdfscreen`
`\usepackage[forpaper]{exerquiz}`

Opcija *preview*

Paket `exerquiz` može generirati različite tipove polja za unos: gumb(buttons), check boxes, radio buttons i polja za unos teksta. To su PDF objekti i ne mogu se vidjeti u

`dvi` pregledniku. Koristeći `preview` opciju, pravokutnici koji ograničavaju polja za unos omogućuju da se vide polja za unos.
Ova opcija pomaže da se podesi pozicija polja za unos. Opcija se koristi prilikom izrade kviza a ne prilikom objave. Kad je autor zadovoljan s pozicioniranjem i kad je spremno za objavu, opcija se miče.
Ova opcija nije potrebna pri upotrebi s `pdftex` opcije jer `pdftex` ne proizvodi `dvi` datoteku.

Opcija *nodljs*

Ako se radi dokument koji će trebati ispisati ili se dokument sastoji samo od vježbi i rješenja (i ne zahtijevaju JavaScript), veličina dokumenta se može značajno smanjiti koristeći `nodljs` opciju.

Opcija *exercisesonly*

Ako se koristi `exercisesonly` okolina, JavaScript nije potreban. Ova opcija je ekvivalentna prethodnoj.

Opcija *debug*

Izrada JavaScript funkcija može biti problematična. Često je korisno ubaciti nekoliko redova JavaScript kôda koje pomažu u istraživanju problema pojedinih funkcija ili skupa funkcija. Primjerice, potrebno je verificirati da se u funkciju predaju ispravni parametri te da je vrijednost koju vraća funckija ispravna. Na taj način se može postići da Acrobat zapisuje vrijednosti u konzolu te se iste vrijednosti mogu prekontrolirati:

```
console.println("Function myFunc");\\
console.println("Parameters: x="x+", y="+y);\\
console.println(""Return Value: retnValue="+retnValue);
```

U ovom kôdu je korištena `console.println()` metoda koja je dostupna samo u Acrobat aplikaciji, ne i u Readeru. Za Reader se može koristiti `app.alert()`, ali ta metoda nije prilagođena za promatranje vrijednosti velikog broja varijabli koje skripta provodi.
Onome tko nema potpuni Acrobat, ova opcija neće biti korisna.

2.4 Okolina *exercise*

U paketu `exerquiz` `solution` okolina se ugnijezdi unutar `exercise` okoline. S ovim okolinama se mogu kreirati pitanja (vježbe) s rješenjima.

Rješenja su napisana u pomoćnu datoteku `\jobname.sol`. `Hypertext` poveznica je napravljena da poveže vježbu s rješenjem.
Može se definirati i vježba s više dijelova s `hypertext` poveznicom na rješenja pojedinačnih dijelova vježbe.
Okolina `exercise` ima svoj brojač (`eqexno`) ali postoji i opcija za druge brojače ili da se brojač uopće ni ne koristi.
Postoji opcija za postavljanje rješenja odmah nakon iznošenja problema. Ovo je korisno kada se želi da rješenje slijedi zadatak.
Konačno, postoji i opcija skrivanja rješenja. Kada se to koristi, rješenja se zakomentiraju i ne pišu se u `\jobname.sol` datoteku.
Postoji i globalna opcija, `nohiddensolutions`. Kada se ponovno generira dokument, rješenja se napišu u `\jobname.sol` datoteku.

Osnovna upotreba

Sintaksa za `exercise` i `solution` okolinu je:

```
\begin{exercise}
Pitanje.
\begin{solution}
Odgovor na pitanje je
.......
.......
\end{solution}
\end{exercise}
```

Evo jedan primjer: PDF dokument izgleda ovako:

EXERCISE 1.
Odredi prvu derivaciju funkcije: $f(x) = x^2(x^3 + x + 1)$.

Solutions to Exercises

Exercise 1. Ovo je umnožak dviju funkcija pa koristimo pravilo za derivaciju umnoška.

$$\begin{aligned} f'(x) = [x^2(x^3 + x + 1]' &= (x^2)'(x^3 + x + 1) + x^2 \cdot (x^3 + x + 1)' \\ &= 2x(x^3 + x + 1) + x^2(3x^2 + 1) \\ &= 5x^4 + 3x^2 + 2x \end{aligned}$$

□

Kôd za to je:

```
\begin{exercise}

Odredi prvu derivaciju funkcije: $f(x)=x^2(x^3+x+1)$.
\begin{solution}
Ovo je umnožak dviju funkcija
pa koristimo pravilo za derivaciju umnoška.\\
\begin{eqnarray*}
f'(x)=[x^2(x^3+x+1]' & =
& (x^2)'(x^3+x+1)+x^2\cdot (x^3+x+1)'\\
& = & 2x(x^3+x+1)+x^2(3x^2+1)\\
& = & 5x^4+3x^2+2x
\end{eqnarray*}
\end{solution}
\end{exercise}
```

Pitanja i odgovori stoje zajedno. Odgovori su zapisani u datoteku `\jobname.sol`.
Brojač se može redefinirati. U preambulu se stavlja kôd:
`renewcommand{\theeqexno}{\thesection.\arabic{eqexno}}`
i prethodna vježba ima broj ovog poglavlja.
Uobičajene "cross-reference" u LaTeX-u rade kao i inače.
"Exercise" je napisano zelenom bojom u PDF-u ako je to i "Hypertext" poveznica, inače

je napisano plavom bojom. To neće biti link kada se koristi `exercise with parts` ili ako se koristi opcija `nosolutions`. Naravno, s opcijom `forpaper` boja se isključuje i napisano je crnom bojom.
Postoji opcijski argument za `solution` okolinu. Kad LaTeXtraži taj argument, a on ne postoji, makroi počinju tražiti otvorenu uglatu zagradu. Dolazi do problema ako rješenje počinje s matematičkim izrazom. Da bi se to spriječilo,treba prije matematičke okoline staviti naredbu: `relax`.

Opcije *exercise* okoline

Vertikalni razmak umjesto rješenja

Okolina `exercise` se može koristiti za razne testove. Na primjer, ako se želi postaviti pitanje i ostaviti prostor u koji bi učenik upisao odgovor.
Okolina `solutions` ima opcijski parametar za ubacivanje vertikalnog razmaka. Kôd izgleda ovako:

```
\begin{exercise}
Ovo je pitanje.
\begin{solution}[1lin]     %<-- opcijski vertikalni razmak
Ovo je odgovor.
\end{solution}
\end{exercise}
```

Ovakav razmak se pojavljuje samo kad se koristi opcija `nosolutions`.
Kada se sastavlja test za učenike, treba napisati cijeli kôd zajedno s rješenjima pa objaviti s opcijom `nosolutions`. Kasnije se može objaviti ključ s `solutionsafter` opcijom. Ako se obje opcije `solutionsafter` i `nosolutions` pojavljuju, opcija `solutionsafter` poništava opciju `nosolutions`.

Skrivanje nekih rješenja

Neka od rješenja mogu biti skrivena s opcijom `h`. Kôd s ovom opcijom za prethodni primjer izgleda ovako:

```
\begin{exercise}[h]    %rješenje se skriva

Odredi prvu derivaciju funkcije:
$f(x)=x^2(x^3+x+1)$.
\begin{solution}
Ovo je umnožak dviju funkcija
```

```
pa koristimo pravilo za derivaciju umnoška.\\
\begin{eqnarray*}
f'(x)=[x^2(x^3+x+1]' & =
& (x^2)'(x^3+x+1)+x^2\cdot (x^3+x+1)'\\
& = & 2x(x^3+x+1)+x^2(3x^2+1)\\
& = & 5x^4+3x^2+2x
\end{eqnarray*}
\end{solution}
\end{exercise}
```

PDF izgleda ovako:

EXERCISE 1.
Odredi prvu derivaciju funkcije: $f(x) = x^2(x^3 + x + 1)$.

Solutions to Exercises

U PDF-u se vidi da nema više hypertext poveznice na rješenja.

Opcija *nohiddensolutions*

Skrivena rješenja mogu biti uključena u dokument tako da se ukloni naredba `h` na svim mjestima i ponovno kreira PDF ili da se uključi opcija `nohiddensolutions`. Ta opcija poništava opciju `h`.
Skrivena rješenja se otkrivaju i s pozivanjem opcije `solutionsafter`. U ovom slučaju, ako želimo da rješenja ostanu skrivena, koristimo opciju `H` umjesto `h`.

Opcija *noHiddensolutions*

Opciju `H` poništava samo `noHiddensolutions`. U preambulu se upisuje sljedeće:
`\usepackage[noHiddensolutions]{exerquiz}`.
Opcija poništava opciju `h` u cijelom dokumentu.

Opcija *nosolutions*

Ponekad učitelji žele postaviti seriju vježbi na internet, ali ne žele odmah dati i rješenja nego će ih objaviti kasnije. Za to postoji opcija `nosolutions`. Dio preambule izgleda

ovako:

```
\documentclass{article}
\usepackage[pdftex]{web}
\usepackage[nosolutions]{exerquiz}
```

Brojač u *exercise* okolini

Brojač se uključuje naredbom `eqexno` i on broji vježbe u dokumentu. Ako želimo da se brojač resetira nakon svakog poglavlja, u preambulu se stavlja sljedeći dio:

```
\makeatletter
\@addtoreset{eqexno}{section}
\makeatother
```

Opcija *noquizsolutions*

Za online kvizove često može biti nezgodno ako ubacimo odmah i rješenja. Takve stvari se rješavaju uz pomoć opcije verb;noquizsolutions.

Opcija *solutionsafter*

Ovisno o tome gdje želimo da budu rješenja, koristimo opciju `solutionsafter`.
Ako u preambulu stavimo sljedeće:

```
\documentclass{article}
\usepackage[dvipsone]{web}
\usepackage[solutionsafter]{exerquiz}
```

rješenja se pojavljuju ispod pitanja.

2.5 Okolina *shortquiz*

Okolina `shortquiz` se koristi za kreiranje kvizova koji za odgovore imaju višestruki izbor ili nadopunu matematičkim tekstom. Diskusija o matematičkim i tekstualnim nadopunama će biti obrađena kasnije. Okolina dozvoljava prilagođavanje izgleda kviza.

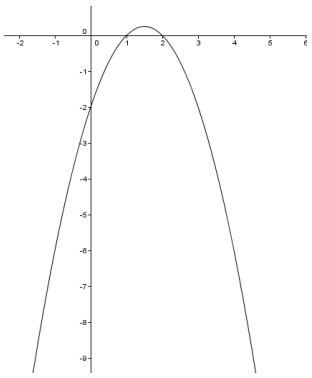
Osnovna upotreba

Sintaksa ove okoline je sljedeća:

```
\begin{shortquiz}          % počinje shortquiz
...Tu idu pitanja...
\begin{answers}{num_col}   tu počinju ponuđeni odgovori
...
\Ans0 ⟨krivi odgovor⟩ &    %krivi odgovor
...
\Ans1 ⟨točan odgovor⟩ &    %pravi odgovor
...
\end{answers}              %kraj ponuđenih odgovora
\end{shorquiz}             %kraj shortquiza
```

Parametar `num_col` je broj stupaca, tj. broj mogućnosti izbora kod zadataka višestrukog izbora. Okolina postavlja `tabular` okolinu ako je `num_col` veći od 1, a `list` okolinu ako je 1.
Ovaj tip kviza je pogodan za kraće serije pitanja, pogotovo odmah nakon objašnjavanja nekog pojma.
Okolina `answers` može centrirati i posložiti odgovore u stupce ukoliko su oni predugi.

Primjer konkretnog kôda za pitanje:
Graf koje funkcije je prikazan na sljedećoj slici?

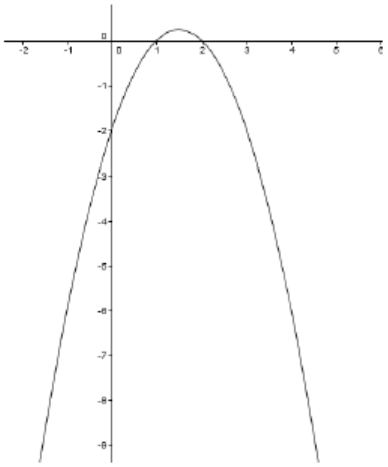


```
\begin{shortquiz}
Graf koje funkcije je prikazan na sljedećoj slici?

\begin{center}
\includegraphics[width=4cm]{2.png}
\end{center}
```

```
\begin{answers}{1}

\Ans{1}$f(x)=\frac{1-x}{x-2}$,\eAns
\Ans{0}$f(x)=\frac{x^2+x}{x-2}$,\eAns
\Ans{0}$f(x)=x^3\sqrt{2-x}$,\eAns
\Ans{0}$f(x)=x+e^{-x}$.\eAns


\end{answers}
\end{shortquiz}
```

Izlaz je ovaj:

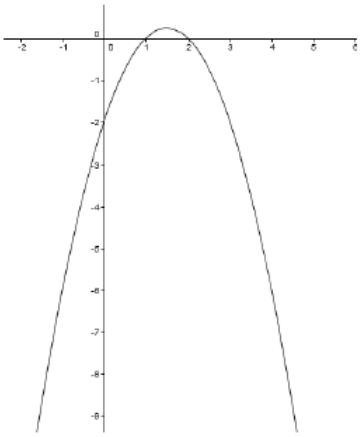
Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = \frac{1-x}{x-2}$,
(b) $f(x) = \frac{x^2+x}{x-2}$,
(c) $f(x) = x^3\sqrt{2-x}$,
(d) $f(x) = x + e^{-x}$.

Ako promijenimo parametar 1 u 2 kod okoline `answers` u prethodnom primjeru, imamo ovaj izlaz:

Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = \frac{1-x}{x-2}$,
(b) $f(x) = \frac{x^2+x}{x-2}$,
(c) $f(x) = x^3\sqrt{2-x}$,
(d) $f(x) = x + e^{-x}$.

Okolina *shortquiz* s radio buttonima

Mali kvizovi sa zadatcima višestrukog izbora mogu biti napravljeni s *radio buttonima*, ne moraju se uvijek koristiti slova ispred odgovora. Za to se koristi `shortquiz` okolina s opcijskim argumentom čija vrijednost je jedinstven naziv za *radio button*. Ukoliko se taj argument ne koristi, naziv će biti automatski dodijeljen.
Na primjer, sljedeći kôd:

```
\begin{shortquiz*}[eksponencijalna]
Graf funkcije $f(x)=a^x$ siječe os $y$ u točki (0,1).
\begin{answers}{4}
\Ans1 Točno & \Ans0 Netočno
\end{answers}
\end{shortquiz*}
```

kad se prevede u PDF dokument izgleda ovako:

Quiz Graf funkcije $f(x) = a^x$ siječe os y u točki (0,1).

☐ Točno ☐ Netočno

Okolina shortquiz s rješenjima

Još jedan tip kviza koji se lako implementira u PDF je kviz s višestrukim izborom i rješenjima koja odmah slijede pitanja. Sintaksa je sljedeća:

```
\begin{shortquiz}
...Tu ide pitanje...
\begin{answers}[⟨ name⟩]⟨num_cols⟩
...
\Ans0 ⟨ krivi odgovor⟩ &
...
Ans1 ⟨ točan odgovor⟩ &
...
\end{answers}
\begin{solution}
 ...Točan odgovor je...
\end{solution}
\end{shortquiz}
```

⟨name⟩ je ime poveznice na rješenje.

Ukoliko kviz izgleda ovako:

Quiz Je li funkcija $f(x) = x \sin x$ derivabilna?

(a) Točno (b) Netočno

Solutions to Quizzes

Solution to Quiz: Odgovor je "Točno". Derivacija je:

$$f'(x) = \sin x + x \cos x.$$

kôd je sljedeći:

```
\begin{shortquiz}
Je li funkcija $f(x)=x\sin{x}$ derivabilna?
\begin{answers}[eksponencijalna]{4}
\Ans1 Točno & \Ans0 Netočno
\end{answers}
\begin{solution}
Odgovor je "Točno". Derivacija je:
$$f'(x)=\sin{x} +x\cos{x}.$$
\end{solution}
\end{shortquiz}
```

Naredbe *bChoices* i *eChoices*

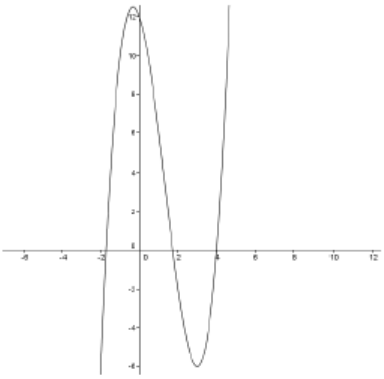
S verzijom 6.03 `exerquiza` par naredbi `bChoices` i `eChoices` je definiran da pomogne složiti listu izbora za pitanja s višestrukim izborom kao odgovorom. Ovaj par makroa nije prava okolina, `eChoices` ustvari i ne radi ništa nego samo glumi kraj oznake. Postoje dva obrazloženja za to:

1. Da bi se pružio prikladan način za popis drugih izbora za odgovore višestrukog izbora, "okolina" omogućava da se jednostavno promijeni broj kolumni ili da se ide iz tablice u popis i obrnuto.

2. Dane su alternative na jedinstven način što olakšava razvoj tehnika za njihov slučajan izbor.

Na ekranu vidimo pitanje:

Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = x^3 - 3x + 2$, (b) $f(x) = (x^2 + x)(x - 2)$,
(c) $f(x) = x^3 - \frac{3}{2}x^2$, (d) $f(x) = x^3 - 4x^2$.

Kôd za to pitanje je:

```
\begin{answers}{2}                          %2 stupca
\bChoices
\Ans{0}$f(x)=x^3-3x+2$,\eAns                %izbor se nalazi između
\Ans{0}$f(x)=(x^2+x)(x-2)$,\eAns            %\Ans...\Ans
\Ans{0}$f(x)=x^3-\frac{3}{2}x^2$,\eAns
\Ans{1}$f(x)=x^3-4x^2 $.\eAns
\eChoices
\end{answers}
```

Ponuđeni odgovori počinju s `\Ans` makroom, a nakon toga slijedi 0 ili 1, što označava koji je točan odgovor. Kraj ponuđenih odgovora je označen s `\eAns` naredbom koja se koristi da se ograniče argumenti. Nije greška ako se ne stavi znak za kraj kolumne "`&`" ili znak za novi red "`\\`" jer to radi naredba `\bChoices`.
Naredbe `\bChoices` i `\eChoices` se mogu koristiti i unutar `answers` okoline kao dio `shortquiz` ili `quiz` okoline.
Da smo, na primjer, u gornjem primjeru argument kod `answers` okoline stavili veći broj,

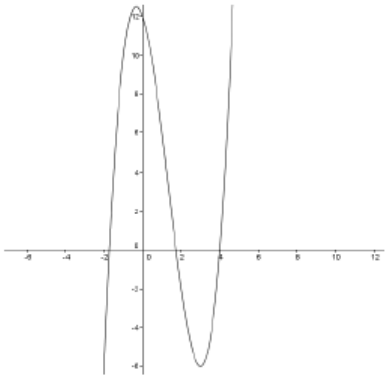
značilo bi da koristimo `tabular` okolinu s tim brojem stupaca. Dalje se može kod `\bChoices` u uglate zagrade [] navesti broj manji od argumenta kod `answers` okoline i to bi značilo da će biti samo toliko kolumni. Kad se ne unese taj argument, ponuđeni odgovori su raspoređeni u prvotno naveden broj kolumni.
Kada je argument kod `answers` 1, opcijski argument kod `\bChoices` se ignorira a ponuđeni odgovori su složeni u listu.
Kôd je:

```
\begin{answers}{1}

\bChoices
\Ans{1}$f(x)=\frac{1-x}{x-2}$,\eAns
\Ans{0}$f(x)=\frac{x^2+x}{x-2}$,\eAns
\Ans{0}$f(x)=x^3\sqrt{2-x}$,\eAns
\Ans{0}$f(x)=x+e^{-x}$.\eAns
\eChoices

\end{answers}

\verb;\begin{answers}{2}; & \verb;% 2 stupca;\\
\verb;\bChoices; & \\
\verb;\Ans{0}$f(x)=x^3-3x+2$,\eAns; & \verb;
%izbor se nalazi između; \\
\verb;\Ans{0}$f(x)=(x^2+x)(x-2)$,\eAns; & \verb;
%\Ans...\Ans;\\
\verb;\Ans{0}$f(x)=x^3-\frac{3}{2}x^2$,\eAns; & \\
\verb;\Ans{1}$f(x)=x^3-4x^2 $.\eAns; & \\
\verb;\eChoices; & \\
\verb;\end{answers}; & \\

\end{shortquiz}
```

I na kraju, to izgleda ovako:

Pocetak kviza **Tok funkcije**

1. Graf koje funkcije je prikazan na sljedećoj slici?



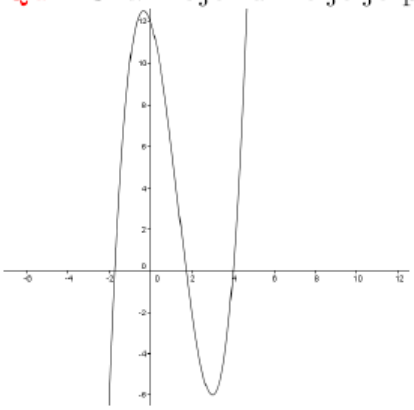
(a) $f(x) = x^3 - 3x + 2$,
(b) $f(x) = (x^2 + x)(x - 2)$,
(c) $f(x) = x^3 - \frac{3}{2}x^2$,
(d) $f(x) = x^3 - 4x^2$.

Mijenjanjem argumenata, autor lako bira izgled ponuđenih odgovora.
Prored (ili razmak između popisa u listi odgovora) se može podesiti s `\rowsep` naredbom.

Ako se stavi `\rowsep{3pt}`, dodat će se još 3pt između redova. Zadana vrijednost (default) je 0pt. Ova naredba se ne smije koristiti unutar `\bChoices/\eChoices`, najbolje je kada se stavi neposredno prije `\bChoices`. Na kraju svake liste taj razmak se postavlja na zadanu vrijednost. Da bi se dobio jedinstveni prored u cijelom dokumentu, potrebno je redefinirati `\rowsepDelault` u preambuli.

S proredom pitanja izgledaju ovako:

Quiz Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = x^3 - 3x + 2$,
(b) $f(x) = (x^2 + x)(x - 2)$,
(c) $f(x) = x^3 - \frac{3}{2}x^2$,
(d) $f(x) = x^3 - 4x^2$.

question okolina

Ova okolina je napravljena da radi zajedno s `quiz` okolinom koja će biti obrađena u sljedećem poglavlju, ali isto tako jednako dobro radi i sa `shortquiz`.
Koristeći ovu okolinu, kvizovi koji su definirani kao `shortquiz` s ili bez rješenja mogu se pomiješati i grupirati da tvore ”mali kviz”.

Opcije *shortquiz* okoline

- `forpaper` opcija
  Ova opcija je već ranije opisana. Odgovori na pitanja u ovoj okolini nisu pisani na posebnim stranicama nego se odvajau s `\medskip`.
  Ako se želi napraviti serija pitanja s višestrukim izborom s rješenjima, mora se imenovati mnogo destinacija(opcijski argument u `answers` okolini). Drugi način je da LaTex sam dodijeli imena i to jedinstvena imena i za to se koristi `\thequestionno`:

```
\begin{shortquiz} Odgovori na pitanje a onda pogledaj odgovore.
 \begin{questions}
   \item
     \begin{answers}[quiz:\thequestionno]{4}
      ....
```

```
    \end{answers}
     \begin{solution}
      ...
       \end{solution}
\end{questions}
\end{shortquiz}
```

- `solutionsafter` opcija
  Opcija samo postavlja logičku vrijednost. Promjena logičke vrijednosti može biti lokalno kontrolirana sa `\SolutionsAfter` i sa `SolutionsAtEnd`. Prva naredba se stavlja prije `shortquiz` okoline a druga odmah poslije.
  Pitanja i odgovori mogu biti napravljeni zajedno, ali da se prvo objave pitanja, a kasnije pitanja s odgovorima.

- `proofing` opcija
  Opcija `proofing` se koristi za korekturu, a u preambulu se stavlja naredba:
  `\usepackage[proofing]{exerquiz}`
  Simbol koji je definiran s naredbom `proofingsymbol` će označiti točne odgovore kao što je definirano u izvornom kôdu. Naredba `proofingsymbol` se može redefinirati, a definicija je
  `\newcommand\proofingsymbol{\textcolor{webgreen}}{$\bullet$}`
  Ova opcija vrijedi i za `quiz` okolinu.

- `showgrayletters` opcija
  Ako se koristi ova opcija, slova A, B, C, itd. ispred ponuđenih odgovora će biti siva. Upotreba ove opcije je globalna i kontrolirana s prekidačem `ifaebshowgrayletters`. Siva slova se mogu uključiti i isključiti lokalno. Da bi se ovo omogućilo, treba ubaciti naredbu `aebshowgrayletterstrue`. Za isključivanje se koristi naredba `aebshowgraylettersfalse`.

- Micanje rješenja
  Set rješenja uglavnom dolazi na kraju dokumenta. Ta pozicija se može seliti ako se uključi naredba `includequizsolutions` u bilo kojem trenutku nakon zadnjeg zadatka.

Redizajn *shorquiz* okoline

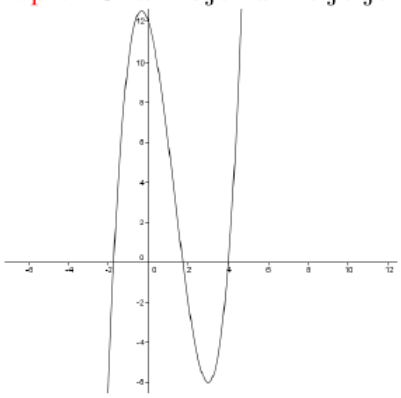
Izgled kviza se može prilagoditi svakom autoru. Evo kako se mogu promijeniti naslovi i razni obrasci.

- Mijenjanje naslova
  Može se privremeno promijeniti naslov za `shortquiz` okolinu redefiniranjem makroa `\sqlabel`. Sintaksa za redefiniranje je:
  `\renewcommand\sqlabel{...novi kôd...}`
  Može se promijeniti i zadana oznaka. Ako nema `\sqlabel`, onda `shortquiz` koristi zadani naslov. Zadani naslov je `\eq@label` i mora se redefinirati koristeći makro `\renewcommand`. Najbolje je da se ovo napravi u preambuli.
  Da bi se u cijelom dokumentu umjesto Kviz nalazilo Ispit, potrebno je napraviti sljedeće izmjene u preambuli:

```
\makeatletter
% promijeni naziv "Kviz" u "Ispit"
\renewcommand\eq@sqlabel{\textcolor{red}{Ispit.}}
% promijeni oznaku za davanje rješenja kviza
\renewcommand\eq@sqslrtnlabel{Kraj ispita}
%promijeni oznaku za rješenja
\renewcommand\eq@sqsllabel{%
   \string\textbf{Rješenja ispita:}}
\renewcommand\eq@sqslsectitle{Rješenja ispita}
%promijeni zadano zaglavlje za rješenja
\renewcommand\eq@qslsecrunhead{Rješenja ispita}
\maketother
```

Pitanje sada izgleda ovako:

Ispit. Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = x^3 - 3x + 2$, (b) $f(x) = (x^2 + x)(x - 2)$,
(c) $f(x) = x^3 - \frac{3}{2}x^2$, (d) $f(x) = x^3 - 4x^2$.

- Modificiranje obrazaca
  Izgled radio buttona se može modificirati koristeći naredbu `\everysqRadioButton`. Osim njih, izgled se može mijenjati i svim drugim poljima za unos ili odabir odgovora.

  Na slici je jedan primjer. Gornja polovina slike je napravljena s zadanim (defaultnim) naredbama, a donja s izmjenjenim.

Quiz Graf funkcije $f(x) = a^x$ siječe os y u točki $(0,1)$.
☐ Točno ☐ Netočno

Quiz Graf funkcije $f(x) = a^x$ siječe os y u točki $(0,1)$.
☐ Točno ☐ Netočno

2.6 Okolina *quiz*

Koristeći ovu okolinu radimo kvizove koji se odmah ocjenjuju. Na primjer, nekoliko pitanja je grupirano u jedan kviz. Učenik ga riješi a JavaScript sprema te odgovore. Nakon što završi kviz, prikaže mu se broj bodova koje je postigao.
`quiz` okolina može generirati odgovore s višestrukim izborom i odgovore koje treba nado-

puniti s tekstom ili matematičkim izrazom.
Dva su tipa kviza, `link-style` i `form-style`. Kasnije ćemo vidjeti da ova okolina može i ispravljati kvizove.
Okolina se sastoji od serije ugniježđenih okolina. Unutar `quiz` okoline nalazi se `question` okolina a unutar nje je `answers` okolina.
Bilo gdje u dokumentu se stavi makro `\ScoreField` da bi se prikazali rezultati kviza. Važno je da se vrijednost parametra makroa podudara s `quizfieldname` koji je definiran u argumentu `quiz` okoline.

Osnovna upotreba

Učeniku se moraju dati upute da bi se sve dobro napravilo. Prvo se mora kliknuti na "Početak kviza" da bi se on inicijalizirao. Ako se to ne napravi, odmah se javi poruka o grešci. Na kraju se klikne na "Kraj kviza". Za vrijeme rješavanja kviza a prije klika na "Kraj kviza", odgovori se mogu nebrojeno puta izmijeniti. Pojavi se prozor s porukom s prošlim odgovorom i pita jesmo li sigurni da želimo promijeniti odgovor.

Form Buttons za početak i kraj kviza
Kviz rađen u `quiz` okolini mora imati hypertext poveznicu za "Početak kviza" i "Kraj kviza". Sintaksa je sljedeća:

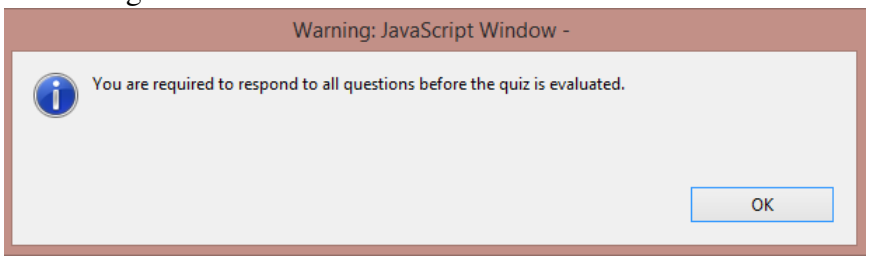
```
\useBeginQuizButton
\useEndQuizButton
```

Obje ove naredbe imaju opcijski argument koji se može koristiti za modificiranje njihovog izgleda.
Postavljanje praga

U `quiz` okolini je zadano da učenik može započeti kviz i završiti da ne odgovori ni na jedno pitanje. To se zove `lowThreshold` i to je zadano ponašanje (default behavior).
Autor kviza može postaviti `\highThreshold` redefinirajući `minQuizResp` makro. Zadana definicija je `\newcommand\minQuizResp{lowThreshold}`

Ako se stavi `\renewcommand\minQuizResp{highThreshold}`, učenik onda mora odgovoriti na sva pitanja.
`\lowThreshold` i `\highThreshold` su JavaScript funkcije koje se pozivaju kad se klikne na "Kraj kviza". Ako se prag ne spomene, pojavit će se poruka koja upozorava na to. Autor kviza može napisati dodatno prilagođenu funkciju za prag i staviti njeno ime u `\minQuizResp` makro.

Poruka izgleda ovako:



Ispravljanje odgovora s JavaScriptom

S verzijom exerquiza 1.2 mogu se ispravljati kvizovi napravljeni u `quiz` okolini. Da bi se to napravilo, treba ubaciti još i gumb za ispravljanje. On se instalira koristeći makro `\eqButton`.
Makro `\eqButton` radi lijepi znak za gumb za ispravljanje. JavaScript ispravlja kviz. Jedini argument koji se zahtijeva tu je oznaka polja koja jedinstveno definira polje u koje se upisuje konačan zbroj. "Correction" gumb je definiran s 2 parametra.
`\eqButton[mod_appear]{field_name}`

Drugi parametar je ime polja koje sadrži konačan zbroj bodova u kvizu. Ima i opcijski argument koji se odnosi na modificiranje izgleda gumba. Osim toga, postoji i globalni mehanizam za modificiranje izgleda. To su globalne modifikacije: `\everyButtonField` i `everyeqButton`. Prvi mijenja izgled svakog gumba u kvizu a drugi sva `\eqButtons`.
`\eqButton` neće raditi dok se ne klikne na "Kraj kviza". Korisnik može ponovno riješiti kviz jednostavno ako klikne na "Početak kviza" a sva polja i JavaScript varijable će se biti čiste.

`nocorrections` opcija

Dodavanje dijela za ispravljanje ubacuje dosta JavaScript kôda u pdf dokument i to je tako zadano. Ako želimo dokument bez opcije koja nudi ispravljanje kviza, samo trebamo specificirati `nocorrections` u listu opcija za exerquiz. Postoji par makroa koji se mogu koristiti da ponište neki izbor, a to su `CorrectionsOn` i `CorrectionsOff`. Svaki od njih funkcionira dok se ne pozove drugi.
Ako se uključi ova opcija, onda se ne pojavljuje gumb `eqButton`.

Kvizovi s rješenjima

Kao što se može ponuditi ispravljanje kviza, tako se mogu dati i rješenja na pitanja. Za to se koristi `solution` okolina i priloži se ime destinacije za tu okolinu.
Nakon što je kviz gotov i pritisnut je gumb za ispravljanje, pojavi se ispravljeno. Točan odgovor ima zeleni rub. Ne moraju sva pitanja imati odgovor. Ponekad u `answers` okolini

se može pojaviti opcijski parametar zapisan u uglate zagrade. Vrijednost tog parametra je jedinstveno ime za rješenje kviza.

Na slici je primjer s točnim i netočnim odgovorom da se vidi na koji način JavaScript boja polja za unos odgovora u ovisnosti o točnosti odgovora.

Oblikovanje *quiz* okoline

Postoji 4 načina kako se može promijeniti izgled kviza:

- promjena naslova
- promjena izgleda check
- promjena polja u kojemu se nalazi rezultat
- promjena izgleda gumba ”Correction”

Može se redefinirati naziv kviza ili bilo koje druge oznake; i to lokalno i globalno.
Lokalno: `\renewcommand\bqlabel{Begin Exam}` `\renewcommand\eqlabel{End Exam}`
Globalno:

```
\makeatletter
\renewcommand\eq@bqlabel{Begin Exam}
\renewcommand\eq@eqlabel{End Exam}
\makeatother
```

Koristeći makro `symbolchoice` u paketu `exerquiz`, mijenja se izgled ”checka”. Dozvoljene vrijednosti argumenta za `\symbolchoice` su check (defaultni), krug, križ, romb, kvadrat i zvijezda.

Boje kojima se nešto označava u kvizu također se mogu mijenjati i to redefiniranjem naredbe `\checkColor`, `\crossColor` i `\correctColor` u preambuli ili prije. Ovo su defaultne naredbe:

```
\renewcommand\checkColor{color.red}
\renewcommand\crossColor{color.red}
\renewcommand\correctColor{["RGB", 0, .6, 0]}  %webgreen
```

Boje se ubacuju koristeći JavaScript.
Polje za bodove je tekstualno polje u koje kviz javlja zbroj. Polje se može napraviti koristeći makro:
`\ScoreFields`: `\ScoreField[mod_appear]{field_name}`

Najjednostavniji slučaj je kada taj makro ima samo jedan argument, a dodavanjem opcijskog parametra može se modificirati izgled polja.

2.7 Pitanja objektivnog tipa

Pitanja objektivnog tipa su se pojavila s `exerquiz` verzijom 2.0 i u ovom poglavlju su objašnjene sve naredbe koje se koriste u njihovom kreiranju.

Matematička i tekstualna pitanja

Exerquiz razlikuje 2 tipa pitanja koji zahtijevaju odgovor duži od da/ne:

1. Matematička pitanja na koja se mora odgovoriti s matematičkim izrazom
2. Pitanje na koje treba odgovoriti tekstualno.

Matematička pitanja

Na matematička pitanja se odgovara funkcijom s jednom ili više varijabli x, y, z, itd. Kad se tim varijablama da neka vrijednost, odgovor se svodi na broj.
Na primjer, treba odgovoriti na pitanje:

$((x-1)^2)' =$ ☐

Odgovor na to pitanje je funkcija jedne varijable.
Kôd za to pitanje izgleda ovako:

```
\begin{shortquiz}[answer] Odredi derivaciju funkcije:

\begin{questions}
```

```
\item $((x-1)^2)'
    =\\ \RespBoxMath{2(x-1)}(xy){2}{.0001}
    {[0,1]x[0,1]&[-2,-1]x[-2,-1]}$\hfill
\CorrAnsButton{2(x-1)}\kern1bp\sqTallyBox


\end{questions}
\end{shortquiz}
```

`\RespBoxMath` naredba se koristi za postavljanje pitanja objektivnog tipa i mora se pojaviti u `shortquiz` i `quiz` okolinama.
Algoritam radi na sljedeći način: učenik unese odgovor i klikne na tipku "Enter". Odgovor koji je unešen i točan odgovor se uspoređuju i ocjenjuju. Unaprijed je određeno u kolikoj mjeri dani odgovor smije odstupati od točnog. Ukoliko se dani odgovor razlikuje od točnog u većoj mjeri od zadane, odgovor se ne priznaje kao točan.
Naredba `\RespBoxMath` ima 10 parametara, 5 opcijskih i 5 obaveznih:
`\RespBoxMath[#1]#2(#3)[#4]#5#6#7#8[#9]*#10`.
Objašnjenje parametara:

- `#1` : Opcijski parametar koji se koristi za modificiranje izgleda tekstualnog polja.
- `#2` : Točan odgovor na pitanje. Mora biti unešena numerička vrijednost ili funkcija jedne varijable. Funkcije kao što su $\sin(x)$ i $\cos(x)$ su vrste matematičkih objekata u JavaScriptu. Ipak nije potrebno tipkati `Math.sin(x)` ili `Math.cos(x)` nego se izraz unese s grupom `with(Math)`. Na primjer;
  `with(Math){ 2*sin(x)*cos(x) }`.
- `#3` : Opcijski parametar, omeđen zagradama, koji definira nezavisnu varijablu. x je zadana (defaultna) vrijednost. Ovaj parametar se uvijek mora odvojiti zagradama.
- `#4` : Opcijski parametar, sadrži ime destinacije rješenja pitanja. Ako se ovaj parametar pojavljuje, rješenje mora ići poslije pitanja priloženo u `solution` okolini.
- `#5` : Broj korišenih samples points, uglavnom su dovoljna 3 ili 4.
- `#6` : Preciznost koja se zahtijeva.
- `#7` : Parametri `#7` i `#8` se koriste za definiranje intervala iz kojeg se uzimaju sample points. Dvije su forme:
  1. `#7` je početak intervala a `#8` kraj, i to je zastarjela upotreba.

2. interval se zapisuje na standardan način: $[a, b]$. Tamo gdje je parametar `#2` lista od više varijabli, intervali za svaki varijablu se odvajaju znakom x, `[0,2]`x`[1,2]`x`[3,4]`.

- `#8` : Dva su načina upotrebe:

  1. `#8` je desni rub intervala, što je zastarjela upotreba.
  2. u ovom slučaju se parametar `#8` ne koristi

- `#9` : Opcijski parametar koji sadrži ime prilagođene funkcije uspoređivanja.

- `#10` : Može se očitati samo ako je ispred njega znak $*$ i sadrži ime JavaScript funkcije koja se koristi da obradi ono što je unešeno za odgovor.

Neke napomene:

- Točan odgovor se može napisati uz pomoć JavaScripta ili onako kako se očekuje od učenika da unese odgovor. Koriste se i funkcije i operatori.

- Interval iz kojeg se uzimaju sample points se mora pažljivo odabrati. Naravno, interval mora biti iz domene funkcije u odgovoru. Mora se paziti da se uzme interval bez ijednog singulariteta.

- JavaScript u Acrobatu 5.0 zna kako raditi s iznimkama, ali to još uvijek nije umetnuto u kôd.

S prvim parametrom se može mijenjati izgled polja lokalno, ali postoji i globalni način:

- `\everyeqTextField` mijenja izgled svakog polja

- `\everyRespBoxMath` mijenja izgled samo onih polja koja su kreirana s naredbom `RespBoxMath`.

Dodatna pojašnjenja

Uključivanje polja "Odgovori" s naredbom *\CorrAnsButton*

Točan odgovor se može uključiti u pitanje, samo je potrebno staviti naredbu `\CorAnsButton`. Naredba ima 1 parametar a to je točan odgovor koji će se vidjeti klikom na gumb i obično je taj odgovor identičan onom koji se nalazi u 2 argumentu (treba paziti da je 1. argument opcijski) u naredbi `\RespBoxMath`.

Primjer 2.7.1. *Odredi derivaciju sljedeće funkcije:*
$(\cos(\frac{1}{x}))' =$

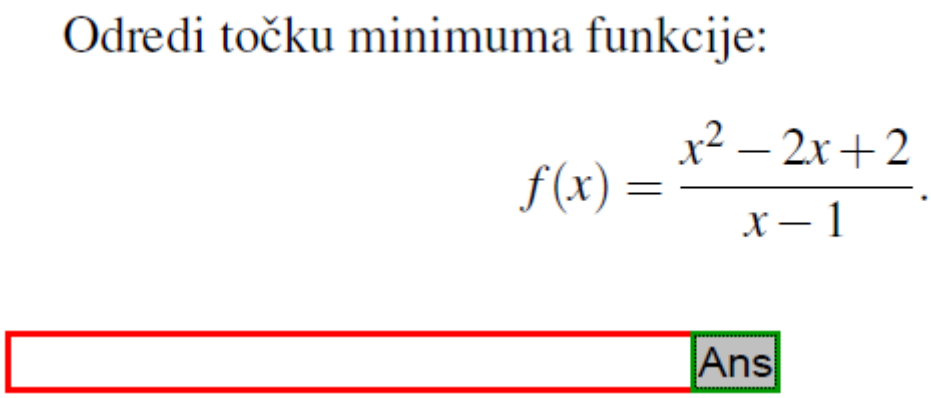
Kôd je sljedeći:

```
\begin{shortquiz}
\begin{questions}
\item $(\cos(\frac{1}{x}))'
    =\\ \RespBoxMath{1/(x^2)*sin(1/x)}(x){3}{.0001}
    {[-1,1]}$
    \hfill
\CorrAnsButton{1/(x^2)*sin(1/x)}
\kern1bp\sqTallyBox
\end{questions}
\end{shortquiz}
```

Uključivanje rješenja

Osim ispravljanja odgovora, može se napravit i da se izbaci rješenje ukoliko se ne zna odgovor. Mora se unijeti četvrti parametar i nakon pitanja mora ići `solution` okolina s odgovorom.

Kôd za pitanje iz kviza:



```
\begin{quiz}
\begin{questions}
Odredi točku minimuma funkcije: $$f(x)=\frac{x^2-2x+2}{x-1}.$$\\
\RespBoxMath{(2,2)}*{2}{.0001}01\CorrAnsButton{(2,2)}

\begin{solution}
$m=(2,2)$
\end{solution}
```

```
\end{questions}
\end{quiz}
```

Uključivanje Tally Box makroa

Makro `sqTallyBox` se koristi za brojanje netočnih odgovora koje je korisnik unio u polje za odgovor.

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

x+2 Ans 1

Kôd:

```
\begin{shortquiz}[answer] Odredi derivacije sljedećih funkcija:
\begin{questions}
\item $((x^2-1)(x^4+x^2+1)+(1-x^2)^3)'
    =\\ \RespBoxMath{12x^3-6x}(x){4}{.0001}{[0,1]}$\hfill
\CorrAnsButton{12x^3-6x}\kern1bp\sqTallyBox
\end{questions}
\end{shortquiz}
```

Tally box se koristi kod `shortquiz` okoline, ali ne i u `quiz` okolini.

Brisanje unešenog odgovora

Ako se stavi naredba `sqClearButton` u `shortquiz` okolinu, mogu se brisati unešeni odgovori. Ta naredba se u kôd stavlja nakon naredbe `sqTallyBox`. Između te dvije naredbe ide naredba `kern1bp` da odvoji polja `sqTallyBox` i `sqClearButton` da im se ne bi poklapala polja.

Okolina *shortquiz*

Objektivna pitanja, svejedno sa ili bez correction boxa, `\corrAnsButton` ili tally boxa `\sqTallyBox`, mogu se miješati s pitanjima koja imaju odgovor s mogućnošću višestrukog izbora.
Rješenja mogu biti uključena koristeći `solution` okolinu. Da bi se dobio odgovor na pitanje, klikne se na polje "Ans".
Kada se unutar `shortquiz` okoline koriste objektivna pitanja, u opcijski argument se mora

upisati jedinstveno ime polja. Taj opcijski argument daje skupu pitanja uobičajeno ime i svi podržani makroi koriste to ime. Imenovana destinacija rješenja je unešena s parametrom `#5` u naredbi `\RespBoxMath`. U ovakvim slučajevima uobičajeno je korištenje ugrađenog makroa `\sqTallyTotal`.
U `shortquiz` okolini se mogu dati i pojedinačna pitanja samo se ne smije unutar nje koristiti `questions` okoline.

Okolina *quiz*

I unutar `quiz` okoline isto mogu biti pomiješana pitanja objektivnog tipa i pitanjima koja imaju odgovor s mogućnošću višestrukog izbora. Kada se postavljaju objektivna pitanja u `quiz` okolini, koristi se `\RespBoxMath` i opcijski uključuje `\CorrAnsButton`.
Makro `sqTallyBox` tu nije potreban jer je evaluacija kviza odgođena do trenutka kada se završi kviz.
Gumbi kreirani s `\CorrAnsButton` su skriveni dok se ne završi kviz i dobiju se bodovi pa zatim klikne na corrections button `eqButton`. `\CorAnsButton` nam ne treba ako nema `eqButton`. Ako postoji odgovor na pitanje, gumb "Ans" je uokviren zelenom bojom. Klikom na taj gumb dobiva se rješenje.
`quiz` okolina zahtijeva ime polja.

Prompt Button

Iako se ustalio ovaj izraz, to nije najbolji opis. Autor kviza može dati i prompt button kao nastavak `\CorrAnsButton`.
U nekim kvizovima, autor može postaviti seriju pitanja gdje odgovor na jedno pitanje ovisi o točnom odgovoru na prethodno pitanje. U tom slučaju, želimo potaknuti učenika da da točan odgovor da bolje može odgovoriti na sljedeće pitanje. `\@PromptQuestion` ne nudi samo odgovor na pitanje nego čini da odgovarajuće polje bude samo čitljivo (read only) da korisnik ne može promijeniti već ponuđeni odgovor.
Korisnik prvo unese odgovor i kad je odgovor zadovoljavajući, može dobiti točne odgovore na sljedeća pitanja.

Grupirana pitanja koja traže odgovor matematičkog tipa i tekstualnog tipa

`exerquiz` definira grupirajuću okolinu, `mathGrp`, za pitanja na koja se dopunjuje matematičkim izrazom i pitanja na koja se odgovor nadopunjuje s tekstom tamo gdje odgovor na pitanja može tražiti da se unese tekst u nekoliko matematičkih polja.
Dok se koristi ova okolina, umjesto gumba `\CorrAnsButton` imamo `\CorrAnsButtonGrp`. Obavezan argument za ovaj gumb je zarezom odvojena lista odgovora koja se pojavljuju

unutar grupiranih pitanja. Ta grupa pitanja se izvodi jednako kao da ima i samo jedno pitanje. Odgovori trebaju biti u istom redoslijedu u kojem TeX procesira matematička pitanja. Ako npr. pitanje ima 3 prazne kućice za odgovor i u jednu se ne ponudi odgovor, `ScoreField` vraća "0 od 1". Da bi odgovor na to pitanje bio točan, u svim kućicama treba biti točan odgovor.
Bodovi mogu biti pridruženi individualnim odgovorima a rezultat je dan na osnovu vrijednosti unešenog i odgovarajućih bodova. Postoji zadana JavaScript funkcija koja zbraja bodove ali autor dokumenta može napraviti prilagođenu funkciju ako želi drugačije bodovanje.

Modificiranje gumba

Svi gumbi imaju prvi opcijski parametar za modificiranje njihovog izgleda, ali postoji i globalna naredba za koa ih isto tako mijenja.

Naredbe za globalnu modifikaciju	
shortquiz okolina	
`\Ans`	`\everysqRadioButton`
`\sqTallyBox`	`\everysqTallyBox`
`\sqTallyTotal`	`\everysqTallyTotal`
`\sqClearButton`	`\everysqClearButton`
quiz okolina	
`\useBeginQuizButton`	`\everyBeginQuizButton`
`\useEndQuizButton`	`\everyEndQuizButton`
`\Ans`	`\everyqRadioButton`
`\ScoreField`	`\everyScoreField`
`\eqbutton`	`\everyeqButton`
`\AnswerField`	`\everyAnswerField`
`\PointsField`	`\everyPointsField`
`\PercentField`	`\everyPercentField`
`\gradeField`	`\everyGradeField`
obje okoline	
`\RespBoxMath`	`\everyRespBoxMath`
`\RespBoxTxt`	`\everyRespBoxTxt`
`\CorrAnsButton`	`\everyCorrAnsButton`

Osim ovih, postoje i druge "every" naredbe koje utječu na izgled raznih gumba i polja za unos teksta. Dvije naredbe `\everyeqButton` i `\everyeqTextField` se izvršavaju prije bilo kojeg `exerquiz` gumba i polja za unos teksta. To se uglavnom koristi za generalni izgled za sva shortquiz ili kviz polja.

2.8 Proširenje s *dljslib* i *insdljs*

Cijeli paket `exerquiz`, a posebno dio s poljima za unos matematičkih odgovora se može dosta programirati. Dva su načina za to:

1. koristeći paket `dljslib` što je JavaScript knjižnica ekstenzija
2. ili ako se napiše vlastita prilagođena ekstenzija koristeći paket `insdljs` za umetanje JavaScripta u PDF dokument.

Paket *dljslib*

Paket `dljslib` je ustvari knjižnica JavaScript funkcija. Za sada postoje kôdovi koji daju odgovore na pitanja na koja se odgovara matematičkim tekstom koja se odnose na vektore i računanje vrijednosti funkcija (equations). Postoji i JavaScript funkcija koja procjenjuje odgovore gdje se pojavljuju neodređeni integrali.

- Vrijednosti: Ispod je dio kôda koji se unosi u preambulu tog dokumenta. Specificira se `\usepackage` naredba za paket `\dljslib` da bi se korstila neka JavaScript funkcija.

```
\documentclass{article}
\usepackage{amsmath, amscd}
\usepackage[tight, pdftex, designi, nodirectory]{web}
\usepackage{exerquiz}
\usepackage[equations]{dljslib} % izabere se equations koji želimo
```

- Vektori: Postoje JavaScript funkcije za obradu pitanja o vektorima. Preambula se može zamijeniti s:

```
\documentclass{article}
\usepackage{amsmath, amscd}
\usepackage[tight, dvipdfm, designi, nodirectory]{web}
\usepackage{exerquiz}
\usepackage[vectors, indefIntegral]{dljslib}
```

Paket *insdljs*

S ovim paketom se mogu pisati vlastite JavaScript funkcije direktno u LaTeX fajl. Prilagođene JavaScript funkcije su ubačene u dijelove PDF-a gdje se i inače nalaze JavaScript funkcije. Ovo je samostalan paket i ne treba exerquiz iako ga exerquiz koristi da ubaci JavaScript funkcije u dokument.

2.9 Postavljanje kviza na web server

Kvizovi kreirani u \quiz okolini su skroz samostalni. Oni funkcioniraju na web pregledbnicima (ili u Adobe Readeru) bez komunikacije s drugim serverima. Ovakav kviz je idealan za motivirane učenike koji mogu/žele pročitati materijal i riješiti kviz.
Da bi se slali odgovori na neki web server, moraju se prvo pohraniti u nekoj bazi podataka.

Tehničke informacije

Sve što se treba napraviti je redefinirati link ili gumb "Kraj kviza" da bi se rješenja postavljala na web server ili CGI, ovisno što se želi. Kako se rješenja automatski boduju, CGI rezultate kviza jednostavno pohranjuje u bazu podataka.

- Redefiniranje gumba "Kraj kviza" Sljedeći kôd je kompatibilan i s eq@EndQuizLink i s eq@EndQuizButton, makroima koji kontroliraju link ili gumb na kraju kviza.

```
if (\minQuizResp(\thequestionno)) {\r\t
    var f= this.getField("ScoreField.\curr@quiz");\r\t\t
    if ( f !=null )\r\t\t\t
        this.getField("ScoreField.\curr@quiz").value
            =(\eq@QuizTotalMsg);\r\t\t
    \eq@submitURL
    resetQuiz("\curr@quiz")\r\t
    }
```

  Kôd je mješavina LaTeX makroa i JavaScript funkcija. Makro \eq@submitURL je inače definiran s \@empty. Autor kviza treba samo propisno redefinirati taj makro. To se može napraviti s Acrobat JavaScript metodom this.submitForm().
  Algoritam je ovakav:

  1. Kôd se pokrene ako je postavljen prag. Tekst makro \curr@quiz sadrži ime trenutnog kviza.
  2. Ako postoji polje ScoreField.\curr@quiz, onda se upiše ime učenika u to polje
  3. U trenutku kada se pozove makro
     \eq@submitURL,
     pozove se DLJS reset\quiz("\curr\@quiz")
     koji postavlja neke vrijednosti nazačenom dijelu kviza.

- Skupljanje ID informacija s `\textField`
  Uobičajene informacije o identitetu učenika bi isto trebale biti unešene. Postavi se tekstualno polje u dokument i ono se mora popuniti. `exerquiz` ima makro `\textField` za to:

```
\newcommand\FirstName[2]{\textField
     [\DV{FIrst Name}\textFont{TiRo}\textSize{10}\textColor{0 0 1 rg}]
     {IdInfo.Name.First}{#1}{#2}}
```

  S ovim kôdom se napravi polje s imenom `"IdInfo.Name.First"` koje ima dva argumenta: visinu i širinu polja koje se želi napraviti.
  `textField` makro ima 4 parametra:
  `\textField[#1]#2#3#4`.
  Prvi je opcijski a može se koristiti za modificiranje izgleda polja. Drugi je ime polja, a zadnja dva su širina i visina koju želimo da polje ima.

- Prikupljanje specifičnih informacija o kvizu `\eqSubmit`
  Osim osobnih podataka, mogu se pohraniti i podatci o tome kakav je to kviz, o čemu se radi te rezultati kviza. `exerquiz` ima za to poseban makro koji se zove `eqSubmit` koji se može koristiti za prikupljanje osnovnih informacija ovog tipa. Kôd je ovaj:

```
\newcommand\databaseName[1]{\def\db@Name{}
\newcommand\tableName[1]{\def\db@Table{#1}\def\db@Table{}
\newcommand\eqCGI[1]{\def\db@Table{#1}}\def\eq@CGI{}
\newcommand\eqSubmit[3]
     {\eqCGI{"#1"}\databaseName{#2}\tableName{#3}}
```

  Prije kviza se može utipkati:

```
\eqSubmit{http://www.myschool.edu/cgi-bin/myCGI.cgi}%
     {Derivacije}{Kratki test}
\begin{quiz*}{3. test} Odgovori na sljedeća pitanja.
\begin{questions}
...
...
\end{questions}
\end{quiz*}\quad\ScoreField\currQuiz\eqButton\currQuiz

\noindent
Odgovori: \AnswerField\currQuiz
```

Bilo koja izmjena `\eq@submitURL` će uključivati neki od ovih tekstualnih parametara:
`\eq@CGI, \db@Name, \db@Table, \curr@quiz`

- Neke varijable koje se podnose (submit)
  Kada se podnese (submit) kviz na server, podnosi se i vrijednost svih polja osim ako nije drugačije definirano. Osim osobnih informacija, želimo i podnešene rezultate testa. Ove varijable su važne za to:

  1. Vrijednost JavaScriot varijable `Score` je broj točnih odgovora,
  2. LaTeX varijabla brojač `thequestionno` sadrži broj svih pitanja u kvizu,
  3. JavaScript red `Responses` sadrži odgovore učenika. Sadržaj ovog retka može se pretvoriti u string odvojen zarezima koristeći metodu `toString()`, `ResponsestoString()`

Bodovanje

Pitanja nemaju uvijek istu težinu, zato se i različito boduju.

1. `\PTs#1` : Ovaj makro ima jedan argument a to je broj bodova koji je dodijeljen trenutnom pitanju. Ova naredba se piše odmah nakon `\item` u `\question` okolini. Na primjer;

   ```
   \item\PTs{2} Onda ide pitanje.....
   ```

2. `\PTsHook#1` : Ovaj makro s jednim argumentom se može koristiti da se slože dodijeljeni bodovi. U argument se piše ono što se dodijeljuje. Vrijednost dodijeljenih bodova trenutnom pitanju s `\PTs` je sadržana u makrou `\eqPTs`

   ```
   \PTsHook{($\eqPTsˆ{\text{pts}}$)}
   ```

3. Postoje još tri naredbe koje kreiraju tekstualna polja za prikaz rezultata kviza s dodijeljenim bodovima:

   - `\PointsField[#1]#2` : Sveukupan broj osvojenih bodova na kvizu se isto prijavljuje. Parametar `#2` je ime kviza.
   - `\PercentField[#1]#2`: Postotak na kvizu. Parametar `#1` je ime kviza.
   - `\GradeField[#1]#2` : Ocjena na kvizu. Parametar `#2` je ime kviza. Vrijednosti koje se nalaze u ovim poljima su određeni makroom `eqGradeScale`.

4. `\eqGradeScale` : Ovaj makro postavlja ocjenu kviza, zadana (defaultna) definicija je:

```
\newcommand\eqGradeScale{"A", [90,100], "B", [80,90],
      "C", [70,80], "D",[60,70],"F",[0,60]}
```

Na ovakav način svaki kviz može dobiti samo jednu ocjenu. Vrijednost `eqGradeScale` je matrica s parnim brojem elemenata. Elementi s neparnim brojevima su ocjene, a parni brojevi su elementi intervala postotaka. Kad postotak padne u određeni interval, dodjeljuje se ocjena.
Naravno, ova naredba se može modificirati. Ako radimo kviz naprimjer za naše učenike, stavit ćemo ocjene uobičajene za naš sustav.

```
\renewcommand\eqGradeScale{%
    "Odličan (5)", [90, 100],
    "Vrlo dobar (4)", [80, 90],
    "Dobar (3)", [70, 80],
    "Dovoljan (2)", [60, 70],
    "Nedovoljan (1)", [0, 60]
}
```

NoPeeking naredba

Ako se u preambulu dokumenta ili prije kviza unese naredba `\NoPeeking`, onda će kviz s odgovorima (rješenjima) biti zaštićen od znatiželjnih očiju.
To znači da će Adobe Reader automatski vratiti na stranicu s pitanjima ako korisnik (učenik) pokuša vidjeti odgovore prije nego on da svoje i pojavit će se upozoravajuća poruka da je gledanje rješenja prije davanja svojih zabranjeno.

2.10 Primjena Acrobat eDucation Bundlea

Derivacije u srednjoj školi

Učenici se s derivacijama prvi puta susreću u 4. razredu gimnazije.
U NOK-u u četvrtom ciklusu pojavljuje se infinitezimalni račun i tamo piše da će učenici:

- izračunati prirast i prosječni prirast tablično zadanih funkcija te jednostavnih formulom zadanih funkcija;

- protumačiti derivaciju funkcije fizikalno (brzina promjene) i geometrijski (koeficijent smjera tangente u točki) te derivirati polinome;
- pomoću derivacije ispitati tok i nacrtati graf polinoma, ponajprije kvadratnoga i kubnoga;
- primijeniti derivaciju pri rješavanju jednostavnih problema.

Nakon što su učenici usvojili limese a prije nego što definiraju pojam neprekinutosti funkcije, susreću se s pojmom prirasta funkcije. Prije nego što krenu na derivacije, detaljnije se upoznaju s prirastom.
Pratit ćemo odgovarajuće definicije, teoreme i postupke iz udžbenika OZNAKA.

Definicija 2.10.1. *Neka je $x_0 \in D_f$ i Δx prirast varijable x. Prirast funkcije f u točki x_0 označavamo s Δy ili $\Delta f(x_0)$, a definiramo na način:*

$$\Delta y = \Delta f(x_0) := y_1 - y_0 = f(x_0 + \Delta x) - f(x_0).$$

Nakon definicije slijedi primjer na kojem se objašnjava kako se računa prirast.
Primjetimo da se prvo računa prirast u konkretnoj točki, a potom u bilo kojoj. Primjenjuje se princip "od konkretnog prema apstraktnom". Ovakva će situacija biti vrlo česta u narednom tekstu.

Primjer 2.10.2. *Neka je $f(x) = 2x^2 + 5$, odredimo prirast Δy funkcije*

- *u točki $x_0 = 1$,*
- *u po volji odabranoj točki $x_0 \in \mathbb{R}$.*

Rješenje:

- *Računamo po definiciji:*

$$\Delta y = f(1 + \Delta x) - f(1) = 2(1 + \Delta x)^2 + 5 - (2 \cdot 1 + 5) = 2(\Delta x)^2 + 4\Delta x.$$

- *Ako je x_0 bilo koja točka iz $\mathbb{R}$, tada koristeći definiciju imamo:*

$$\Delta y = f(x_0 + \Delta x) - f(x_0) = 2(x_0 + \Delta x)^2 + 5 - (2(x_0)^2 + 5) = 2(\Delta x)^2 + 4x_0\Delta x.$$

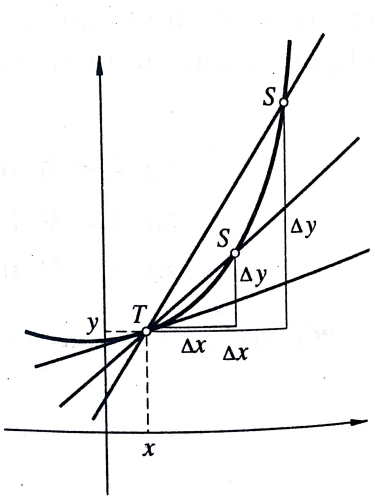
Prirast funkcije koristi se pri određivanju nagiba grafa funkcije. Prvo se trebamo prisjetiti kako se definira nagib pravca, a potom se nagib grafa funkcije definira kao nagib pripadne tangente. Preciznije definicija nagiba funkcije grafa glasi ovako:

Definicija 2.10.3. *Nagib grafa jednak je koeficijentu smjera k, a može se izraziti kao*

$$k = \frac{\Delta y}{\Delta x} = \operatorname{tg} \alpha,$$

pri čemu je α kut što ga pravac zatvara s pozitivnim dijelom x-osi.
Neka se na graf funkcije f može povući tangenta u točki (x_0, y_0). Nagib grafa funkcije f u točki (x_0, y_0) definiramo kao nagib pripadne tangente.

Koeficijent smjera tangente određuje koliki je nagib funkcije u nekoj točki. No, njega nije jednostavno odrediti pa zato tom problemu pristupamo koristeći se sekantom. Sekanta je pravac koji siječe graf funkcije u (barem) dvije točke.



Sekanta će bolje aproksimirati tangentu u točki (x, y) ako je vrijednost prirasta Δx manja. Iz jednadžbe sekante dobivamo jednadžbu tangente kada Δx teži nuli.
Neka je $T(x_0, y_0)$ točka u kojoj računamo nagib, $S(x_0 + \Delta x, y_0 + \Delta y)$ točka na grafu kroz koju vučemo sekantu TS. Nagib sekante je $\frac{\Delta y}{\Delta x}$. Sekanta prelazi u tangentu u točki T kada se S približava točki T po grafu funkcije f.
Na sljedećem primjeru je objašnjeno kako se računa jednadžba tangente.

Primjer 2.10.4. *Neka je $f(x) = x^3$. Nađimo nagib tangente na graf te funkcije u točki* $(2, 8)$.

Rješenje:
Odredit ćemo koeficijent smjera pravca koji prolazi točkom $(2, 8)$ *i točkom* $(2 + \Delta x, f(2 + \Delta x))$*, koja također leži na grafu funkcije. Vrijedi:*

$$f(2 + \Delta x) = (2 + \Delta x)^3 = 8 + 12\Delta x + 6(\Delta x)^2 + (\Delta x)^3,$$

pa je koeficijent smjera

$$k = \frac{\Delta y}{\Delta x} = \frac{f(2 + \Delta x) - f(2)}{\Delta x} = \frac{[8 + 12\Delta x + 6(\Delta x)^2 + (\Delta x)^3] - 8}{\Delta x} = 12 + 6\Delta x + (\Delta x)^2.$$

Koeficijent smjera teži k 12 kad Δx *teži u nulu pa je nagib tangente u točki* $(2, 8)$ *jednak 12.*

Promatranje kvocijenta $\frac{\Delta y}{\Delta x}$ kad Δx teži nuli je ustvari promatranje limesa:

$$\lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}.$$

Ako taj limes postoji, nazivamo ga derivacija funkcije f u točki x_0 i označavamo s $f'(x_0)$.
Treba napomenuti da su u 4. razredu srednje škole učenici u tom trenutku već upoznati s pojmom limesa funkcije te da znaju računati jednostavnije limese.
Iskažimo i preciznu definiciju derivacije.

Definicija 2.10.5. *Derivacija funkcije* f *u točki* x_0 *je broj:*

$$f'(x_0) = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x},$$

ukoliko taj limes postoji.
Taj broj je jednak nagibu k tangente na graf $y = f(x)$ *u točki* (x_0, y_0)*:*

$$f'(x_0) = k = \operatorname{tg} \alpha,$$

α *je kut što ga tangenta zatvara s pozitivnim dijelom x-osi.*
Za funkciju f *kažemo da je derivabilna (diferencijabilna) na intervalu* $\langle a, b \rangle$ *ako u svakoj točki tog intervala postoji derivacija* $f'(x_0)$*. Time je na intervalu* $\langle a, b \langle$ *definirana funkcija* f' *koju nazivamo derivacija funkcije* f*.*

Da bi funkcija imala derivaciju, mora zadovoljavati neke uvjete.

Teorem 2.10.6. *Ako funkcija ima derivaciju u nekoj točki, tada je ona neprekinuta u toj točki.*

Dokaz. Ako funkcija ima derivaciju u točki x, onda postoji limes:

$$\lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}.$$

Budući da nazivnik teži nuli, da bi limes postojao, i brojnik mora težiti nuli. Zato je

$$\lim_{\Delta x \to 0} f(x + \Delta x) = f(x),$$

što znači da je f neprekinuta u točki x. □

Ovo nije dovoljan uvjet za postojanje derivacije. Funkcija koja je neprekinuta ne mora imati derivaciju u toj točki. Slijedi jedan primjer funkcije koja je neprekinuta, a nema derivaciju u nekoj točki. Postoje primjeri funkcija koje su neprekinute,a nemaju derivaciju u niti jednoj točki, ali se u srednjoj školi ne spominju.

Primjer 2.10.7. *Neka je* $f(x) = |3x - 2|$. *Iako je neprekinuta u svakoj točki, ta funkcija nema derivaciju u točki* $x_0 = 3$. *Pokažimo to:*

$$\frac{\Delta y}{\Delta x} = \frac{f(3 + \Delta x) - f(3)}{\Delta x} = \frac{|3(3 + \Delta x - 2| - 7}{\Delta x} = \frac{|9 + 3\Delta x - 2| - 7}{\Delta x} = \frac{|7 + 3\Delta x| - 7}{\Delta x}.$$

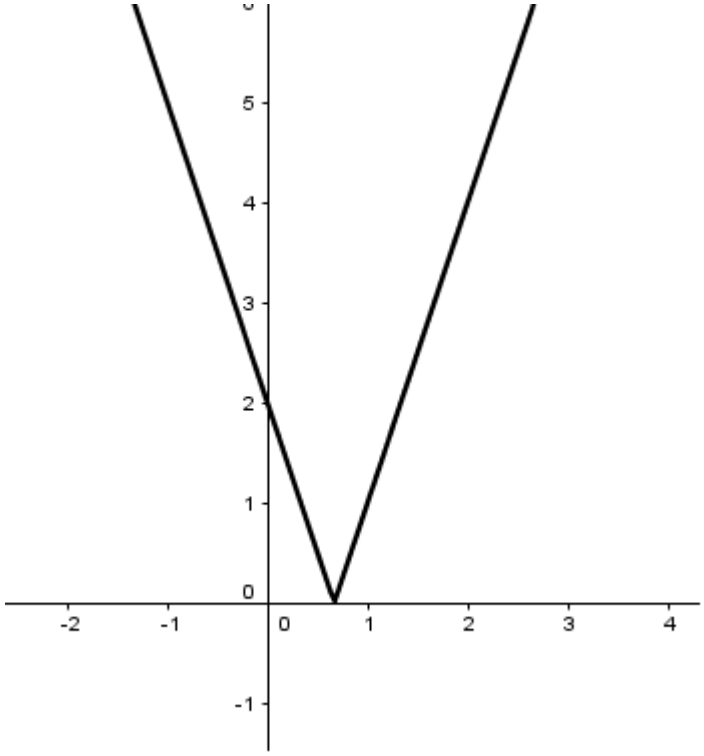
Trebamo odrediti limes kvocijenta $\frac{|\Delta x|}{\Delta x}$ *kad* Δx *teži k* 0. *Ali, ako je* $\Delta x < 0$ *i* $\Delta x \to 0$, *tada je* $\frac{|\Delta x|}{\Delta x} = -3$, *pa je*

$$\lim_{\Delta x \to 0^-} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0^-} \frac{7 - 3\Delta x - 7}{\Delta x} = -3.$$

S druge strane, kad je $\Delta x > 0$ *tada je* $\frac{|\Delta x|}{\Delta x} = 3$, *pa je limes:*

$$\lim_{\Delta x \to 0^+} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0^+} \frac{7 + 3\Delta x - 7}{\Delta x} = 3.$$

Drugim riječima, $\lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x}$ *ne postoji, tj. funkcija nema derivaciju u točki* 3.



Pravila deriviranja

Nakon derivacije, treba pristupiti problemu određivanja derivacije. Učenici u 4. razredu srednje škole izračunavaju derivacije elementarnih funkcija i uče neka osnovna pravila deriviranja. Elementarne funkcije su derivabilne u svakoj točki x iz domene (osim možda u rubnim točkama) pa zato ta točka nije nikako posebno označena.
Sva pravila koja se upotrebljavaju se prvo i dokazuju. Zato je dobro istaknuti koji su bitni koraci za računanje derivacije po definiciji. To su ovi koraci:

1. Izračuna se vrijednost $f(x + \Delta x)$.
2. Izračuna se kvocijent $\frac{\Delta y}{\Delta x} = \frac{f(x+\Delta x)-f(x)}{\Delta x}$.
3. Izračuna se limes $f'(x) = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x}$.

Naravno, što je učenik iskusniji u računanju derivacije, to se ovi koraci stapaju u jedan jedini, tj. računanje po definiciji.
U ovom dijelu nastavne teme dokazuje se niz pravila: pravilo za deriviranje konstantne funkcije, pravilo za deriviranje funkcije $c \cdot f$, linearne, kvadratne i kubne funkcije te funkcije $x \mapsto x^n$, $n \in \mathbb{N}$. Po definiciji se izvode i formule za derivaciju funkcija $x \mapsto \frac{1}{x}$, $x \mapsto \sqrt{x}$, te

pravila za deriviranje zbroja, umnoška i kvocijenta dviju funkcija. Od svih tih pravila dat ćemo detaljan dokaz za derivaciju funkcije $f(x) = \sqrt{x}$.

Primjer 2.10.8. *Odredimo derivaciju funkcije* $f(x) = \sqrt{x}$.
Rješenje: Prvo izračunajmo kvocijent $\frac{\Delta y}{\Delta x}$.

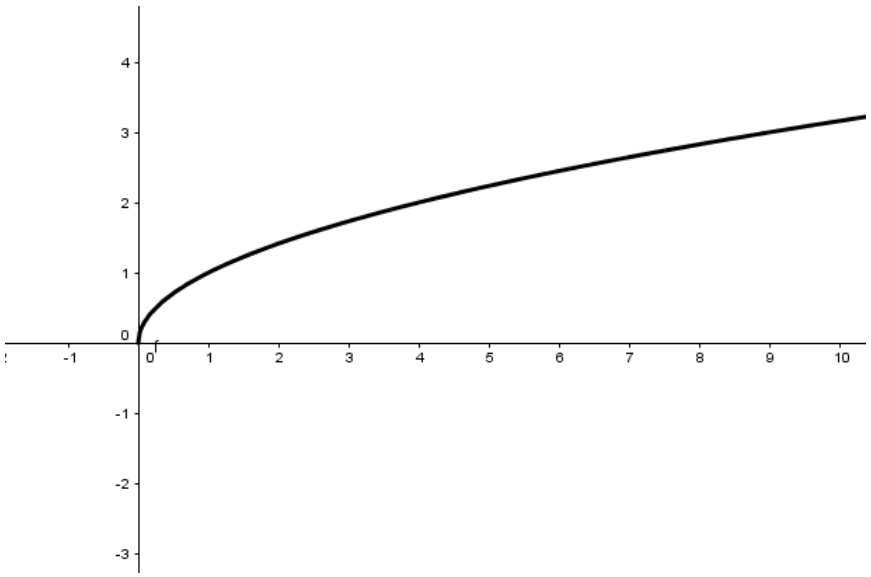
$$\begin{aligned}\frac{\Delta y}{\Delta x} &= \frac{\sqrt{x+\Delta x}-\sqrt{x}}{\Delta x}\\ &= \frac{\sqrt{x+\Delta x}-\sqrt{x}}{\Delta x}\cdot\frac{\sqrt{x+\Delta x}+\sqrt{x}}{\sqrt{x+\Delta x}+\sqrt{x}}\\ &= \frac{(x+\Delta x)-x}{\Delta x(\sqrt{x+\Delta x}+\sqrt{x})} = \frac{1}{\sqrt{x+\Delta x}+\sqrt{x}}\end{aligned}$$

Sad izračunajmo limes:

$$\lim_{\Delta x\to 0}\frac{\Delta y}{\Delta x} = \lim_{\Delta x\to 0}\frac{1}{\sqrt{x+\Delta x}+\sqrt{x}} = \frac{1}{\sqrt{x}+\sqrt{x}} = \frac{1}{2\sqrt{x}}$$

i to je derivacija funkcije f u točki x, $x \neq 0$.

Ovo je prikladan primjer za pokazati jednu situaciju koja se ponekad javlja u proučavanju tijeka funkcije. Funkcija $f(x) = \sqrt{x}$ je definirana na $[0, \infty\rangle$, ali njezina derivacija je definirana samo na $\langle 0, \infty\rangle$. Što se događa u $x = 0$? Pogledajmo pažljivo graf funkcije f u okolini točke $x = 0$.



Kako se ponašaju tangente na graf kad se točka na grafu primiče točki $(0, 0)$?

Tangente postaju sve strmije i teže zauzimanju položaja koji je okomit na $x-$os, tj. kut tangente u $(0,0)$ je $\frac{\pi}{2}$. Naravno, tangens od $\frac{\pi}{2}$ je nedefiniran, ali matematičari često kažu da je nagib takvog pravca ∞.

Dat ćemo još i dokaz pravila za deriviranje umnoška dviju funkcija. Ostala pravila deriviranja se dokazuju slično.

Primjer 2.10.9. *Neka su f i g funkcije derivabilne na intervalu I. Izračunajmo derivaciju funkcije $h = fg$. Stavimo $y = h(x) = f(x)g(x)$. Vrijedi*

$$\begin{aligned}\frac{\Delta y}{\Delta x} &= \frac{h(x+\Delta x)-h(x)}{\Delta x} = \frac{f(x+\Delta x)g(x+\Delta x)-f(x)g(x)}{\Delta x}\\ &= \frac{f(x+\Delta x)g(x+\Delta x)-f(x)g(x+\Delta x)+f(x)g(x+\Delta x)-f(x)g(x)}{\Delta x}\\ &= \frac{f(x+\Delta x)-f(x)}{\Delta x}g(x+\Delta x)+f(x)\frac{g(x+\Delta x)-g(x)}{\Delta x}\end{aligned}$$

Budući da je g derivabilna u x, ona je i neprekidna u x pa je $\lim_{\Delta x\to 0} g(x+\Delta x) = g(x)$, te vrijedi

$$\begin{aligned}\lim_{\Delta x\to 0}\frac{\Delta y}{\Delta x} &= \lim_{\Delta x\to 0}\left(\frac{f(x+\Delta x)-f(x)}{\Delta x}g(x+\Delta x)+f(x)\frac{g(x+\Delta x)-g(x)}{\Delta x}\right)\\ &= f'(x)g(x)+f(x)g'(x).\end{aligned}$$

Primjer kviza

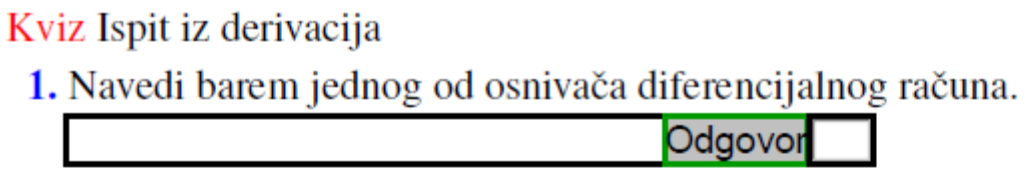
Slijedi primjer kôda u shortquiz okolini.

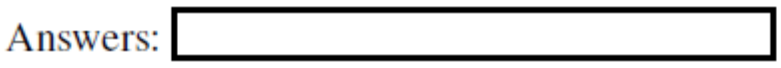
```
\begin{shortquiz}{Ispit iz derivacija}
\begin{questions}


\item Navedi barem jednog od osnivača
diferencijalnog računa.
\par\RespBoxTxt{0}{0}*{5}{Isaac Newton}
{Newton}{I. Newton}%
{Gottfried Leibniz}{Leibniz}%
\CorrAnsButton{Isaac Newton ili
Gottfried Leibniz}
\kern1bp\sqTallyBox
\insAns

\newpage

\item Graf koje funkcije je prikazan
na sljedećoj slici?

\begin{center}
\includegraphics[width=4cm]{1.png}
\end{center}
\begin{answers}{2}

\bChoices
\Ans{0}$f(x)=x^3-3x+2$,\eAns
\Ans{0}$f(x)=(x^2+x)(x-2)$,\eAns
\Ans{0}$f(x)=x^3-\frac{3}{2}x^2$,\eAns
\Ans{1}$f(x)=x^3-4x^2 $.\eAns
\eChoices
\end{answers}\kern1bp\sqTallyBox
\insAns

\newpage

\item
```

```
Funkcija $f(x)=\sqrt{2x^2-x+2}$
nema točku maksimuma.
\begin{answers}{4}
\Ans1 Točno & \Ans0 Netočno
\end{answers}\kern1bp\sqTallyBox
\insAns

\newpage

\item Odredi derivaciju
sljedeće funkcije:\\
 $(x^4-2x^2+3x-1)'=\\
 \RespBoxMath{4x^3-4x+3}*{3}{.0001}01%
\CorrAnsButton{4x^3-4x+3}$
\kern1bp\sqTallyBox
\insAns

\newpage

\item Deriviraj sljedeću funkciju:
$((x-1)^2)'
    =\\ \RespBoxMath{2(x-1)}
    (xy){2}{.0001}
    {[0,1]x[0,1]&[-2,-1]x[-2,-1]}$
    \hfill
\CorrAnsButton{2(x-1)}
\kern1bp\sqTallyBox
\insAns

\newpage

\item Odredi da li funkcija
$f(x)=\sin x\cdot \sin(\frac{\pi}{3}-x)$ u
$x=\frac{\pi}{2}$ raste ili pada.
\par\RespBoxTxt{0}{0}*{3}{padajuća}{pada}{padajuca}%
\CorrAnsButton{pada}\kern1bp\sqTallyBox
\insAns

\newpage
```

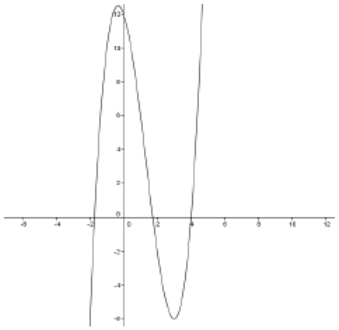
```
\item Odredi točku minimuma sljedeće funkcije:
\\ $f(x)=\frac{x^2-2x+2}{x-1}$,\\
\RespBoxTxt{(2,2)}*{2}{.0001}{03}
\CorrAnsButton{(2,2)}\kern1bp\sqTallyBox
\insAns

\end{questions}
\end{shortquiz}
```

Izlazni dokument je prikazan na sljedećim slikama:





2. Graf koje funkcije je prikazan na sljedećoj slici?



(a) $f(x) = x^3 - 3x + 2$,
(b) $f(x) = (x^2 + x)(x - 2)$,
(c) $f(x) = x^3 - \frac{3}{2}x^2$,
(d) $f(x) = x^3 - 4x^2$.

Answers:

3. Funkcija $f(x) = \sqrt{2x^2 - x + 2}$ nema točku maksimuma.

(a) Točno (b) Netočno ☐

Answers: ☐

4. Odredi derivaciju sljedeće funkcije:
$(x^4 - 2x^2 + 3x - 1)' =$

Odgovor

Answers:

5. Deriviraj sljedeću funkciju: $((x-1)^2)' =$

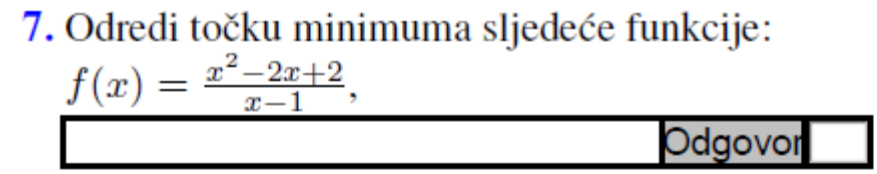
Odgovor

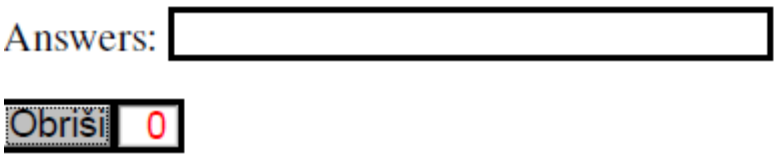
Answers:

6. Odredi da li funkcija $f(x) = \sin x \cdot \sin(\frac{\pi}{3} - x)$ u $x = \frac{\pi}{2}$ raste ili pada.

Odgovor

Answers:

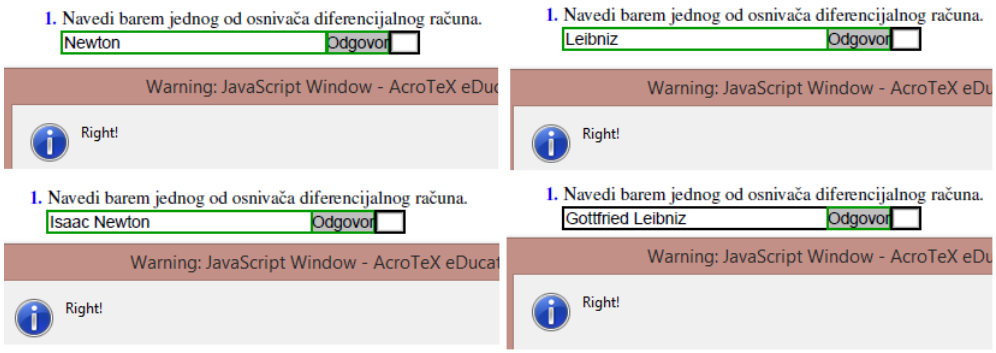




Pogledajmo dio kôda za prvo pitanje:

```
\item Navedi barem jednog od
osnivača diferencijalnog računa.
\par\RespBoxTxt{0}{0}*{5}{Isaac Newton}{Newton}
{I. Newton}%
{Gottfried Leibniz}{Leibniz}%
{Isaac Newton ili Gottfried Leibniz}
```

U naredbi `\RespBoxTxt` 4. su mogući točni odgovori na to pitanje navedeni u zagradama. Ukoliko za odgovor damo bilo koji tekst u zagradi, priznat će se za točno. Dokaz je na sljedećoj slici:

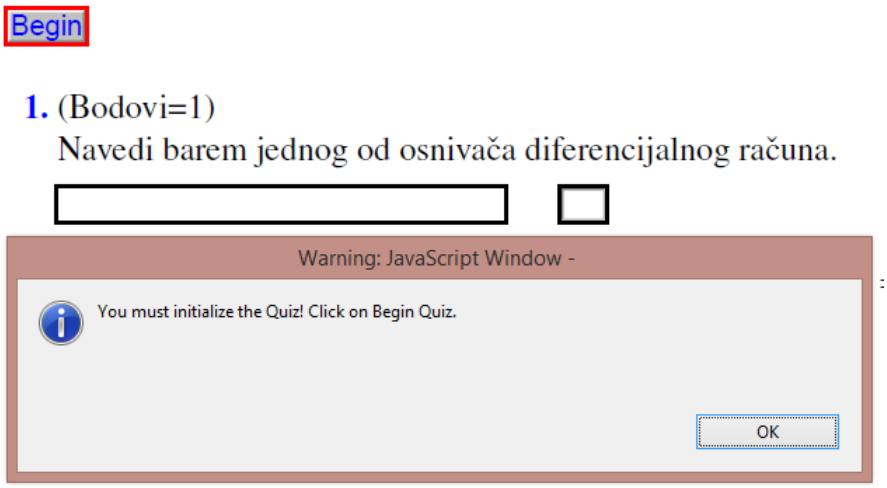


Ako unesemo krivi odgovor, pojavit će se poruka ”Wrong!”.

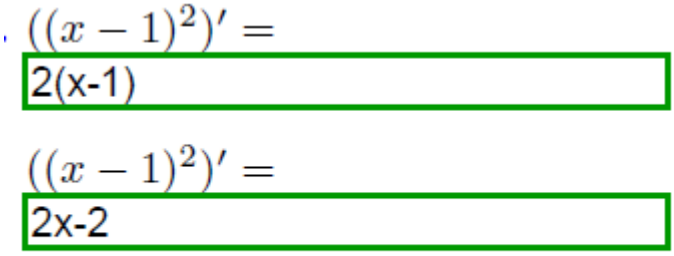
Ako okolinu `shortquiz` zamijenimo s okolinom `quiz` te pitanjima dodijelimo bodove i uključimo naredbe za bodovanje i ocjenjivanje, dobit ćemo ocijenjen ispit.
Nakon što stavimo naredbu za kraj kviza, dodat ćemo ovaj kôd:

```
\end{shortquiz}

\hspace{15mm}\eqButton[\CA{Rezultat testa}]{Ispit iz derivacija}\\
\newpage
\begin{tabular}{ll}
Ispravno odgovoreno na pitanje:& \ScoreField{Ispit iz derivacija}\\
Bodovi:& \PointsField{Ispit iz derivacija}\\
Postotak uspješnosti:& \PercentField{Ispit iz derivacija}\\
Ocjena:& \GradeField{Ispit iz derivacija}
\end{tabular}
\sqClearButton\kern1bp\sqTallyTotal
```

Dogodit će se par promjena u izlaznom dokumentu. Prva promjena je zaglavlje prve stranice, pojavljuje se gumb za početak i ako ne kliknemo prvo na njega nego krenemo odmah pisati odgovor, javit će se poruka o greški.



JavaScript funkcije su nam od velike pomoći kod ispravljanja matematičkih polja. Tako na primjeru 5. zadatku priznaju se 2 odgovora.



Ako na zadatak s višetrukim izborom damo krivi odgovor, prozor će izgledati ovako:

Bibliografija

[1] B. Dakić, N. Elezović, Matematika 2, 2. dio, udžbenik i zbirka zadataka za 2. razred gimnazije, Element, Zagreb, 2009.

[2] B. Dakić, N. Elezović, Matematika 4, udžbenik i zbirka zadataka za 4. razred gimnazije, Element, Zagreb, 2001.

[3] Republika Hrvatska, Ministarstvo znanosti, obrazovanja i športa, Nacionalni okvirni kurikulum, dostupno na `http://www.azoo.hr/images/stories/dokumenti/Nacionalni_okvirni_kurikulum.pdf` (studeni 2015.)

[4] D. P. Story, AcroTeX, eDucational System Tools, dostupno na `http://www.acrotex.net` (2015.)

Sažetak

U ovom radu su obrađena dva paketa AcroTeX softwarea: PDF kalkulator i PDF kviz. Ti paketi su od najveće koristi nastavnicima matematike i sličnih predmeta gdje se točno zna što je točan odgovor.
Rad je koncipiran u dva poglavlja. U prvom je obrađen PDF kalkulator, objašnjene su razne naredbe koje su potrebne za kreiranje dokumenta, zatim je objašnjena tema eksponencijalne funkcije na razini srednjoškolske matematike i dani su primjeri i izgled kalkulatora za izračun vrijednosti eksponencijalne funkcije te zbrajanje i oduzimanje matrica.
U drugom poglavlju je objašnjeno kreiranje PDF kvizova/ispita, dani su kôdovi za razne naredbe, a puno njih ima konkretan primjer sa slikom kako to izgleda u PDF-u. Nakon toga obrađene su derivacije u srednjoj školi i dan je primjer kôda ispita iz derivacija. Dana su dva načina, jedan s ocjenjivanjem, a drugo bez.

Summary

In this thesis, two packages of AcroTeX software, PDF calculator and PDF quiz, have been described. Those packages are very useful for exams with objective style questions.
Thesis consists of two chapters. In each, one package has been processed. PDF calculator is described in first chapter. Various commands have been described and some of them come with sample code. Second chapter is processed in the same way. Beside commands that can be used in those packages, two themes of high schools math have been explained. After all explanations, examples of calculator and quizes are given.

Životopis

Rođena sam 30. studenog 1984. godine u Slavonskom Brodu gdje sam završila osnovnu školu i nakon toga jezičnu gimnaziju. Nakon dugog razmišljanja i trileme, odlučila sam da ću nastaviti školovanje na PMF- Matematički odsjek. Iako sam prvo upisala inžinjerski smjer, 2008. godine sam se odlučila za nastavnički smjer. U međuvremenu sam radila u nekoliko informatičkih firmi. Neki od opisa mojih zaduženja su bili obrada i prikupljanje podataka u nautičkoj industriji, testiranje aplikacija, uređivanje sadržaja za e-knjige koje su se izdavale na engleskom jeziku te sam bila i voditelj tima čija su zaduženja bila prikupljanje podataka te njihov unos u bazu. Paralelno s time, pomagala sam osnovnoškolcima u učenju matematike i tako se odlučila da ipak jednog dana želim poučavati, a ne sjediti u uredu.
Godine 2012. sam upisala diplomski nastavnički studij. Za vrijeme trajanja diplomskog studija postala sam majka dva sina. U slobodno vrijeme volim čitati knjige, kuhati te spremati razne slastice. Osim naše kuhinje, jako volim kinesku i meksičku kuhinju. Trenutno radim u OŠ "Lučko" kao nastavnik matematike.