



Programiranje kroz aplikacije

Uvod

VBE – Programsko okruženje

Procedure

Operatori

Osnovni podaci - osoblje

Predavač: **dr Stefan Vujović**
kabinet: **Laboratorija za obradu signala**
mail: **stefanv@ucg.ac.me**
konsultacije: **svakim danom, po dogovoru**

Saradnik: **Danilo Planinić**
kabinet: **Laboratorija za obradu signala**
mail: **planinic.d@ucg.ac.me**
konsultacije: **svakim danom, po dogovoru**

Osnovni podaci - bodovanje

■ Bodovanje

- 2 kolokvijuma po **25** poena
- Ispit **50** poena
- Izlaskom na popravni kolokvijum/ispit, poništavaju se prethodno osvojeni bodovi i računaju se oni osvojeni na popravnom kolokvijumu/ispitu

■ Ocjene se formiraju po pravilu:

$$50 \leq E < 60$$

$$60 \leq D < 70$$

...

$$90 \leq A \leq 100$$

Literatura

■ Osnovna literatura:

- **Prezentacije sa predavanja**
- **Slobodan Đukanović, “VBA programiranje”, ETF Podgorica, 2011**

■ Dodatna literatura:

Guy Hart-Davis, “VBA6 - Detaljan izvornik”, Kompjuter biblioteka, Čačak, 2002

■ Internet:

<https://docs.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office>

Struktura kursa

- Koncepti Visual Basic for Applications (VBA) programiranja nezavisni od aplikacije (tipovi podataka, operacije, procedure, naredbe za kontrolu toka, nizovi i stringovi) - oko **20%** nastave.
- VBA za MS Excel (uključujući programiranje događaja, VBA forme, rad sa tekstualnim fajlovima i izuzeci) - oko **60%** nastave.
- VBA za MS Word - oko **20%** nastave.

Par osnovnih činjenica o VBA

- VBA je programski (makro) jezik kojim se mogu programirati (automatizovati radnje) aplikacije paketa Office i druge aplikacije.
- VBA predstavlja implementaciju Visual Basic-a.
- VBA je vlasništvo firme Microsoft.

Istorijat

- Prva aplikacija koji je koristila VBA bio je Excel 5. Prije Excel-a 5, Excel je koristio makro jezik XLM.
- U druge aplikacije Office-a, tj. Word, Access, Power Point, Outlook, VBA je ušao u paketu Office 97.
- VBA je bio i sastavni dio FrontPage-a (1997-2003).
- Microsoft preuzima Visio 2000. godine i VBA postaje njegov dio.
- VBA je izašao iz Microsoft Office okvira i biva uključen u Adobe, AutoCAD, CorelDraw, WordPerfect, SolidWorks, ArcGIS.

Istorijat (nastavak)

- Nakon izlaska Office-a 2007, neko vrijeme se spekulisalo o sudbini VBA i mogućnosti da bude zamijenjen .NET jezicima.
- Međutim, zvaničnici Microsoft-a su brzo razjasnili da nije riječ o uklanjanju VBA, već da Microsoft ubuduće neće isporučivati VBA licence.
- Jasno je naglašeno da u Microsoft-u ne nameravaju da uklone VBA iz Windows verzije Office-a.
- Što se tiče Microsoft Office-a za Mac OS, VBA postoji u svim verzijama, osim verzije 2008.
- Sa Office-om 2010, Microsoft uvodi **VBA 7**, koji je dostupan u 32-bitnim i 64-bitnim aplikacijama.
- Poslednja stabilna verzija je **7.1**, uvedena u Office 2019.

VBE – Programsko okruženje

- Za razliku od Visual Basic-a, koji pravi samostalni izvršni fajl, VBA se može izvršavati samo u okviru host aplikacije.
- VBA programski kôd je smješten u samom dokumentu.
- Kod Excel-a 5 i 95, VBA modul se pojavljivao u vidu zasebnog radnog lista u radnoj svesci.
- Počev od Office-a 97, VBA modulima se pristupa koristeći **Visual Basic Editor (VBE)**.
- VBE se startuje prečicom **Alt+F11**.
- Kod aplikacija Office-a 2007 i nadalje, koje koriste traku, VBE se startuje sa **Developer** taba.

VBE – Korisni prozori

- Pri radu sa VBE, korišćićemo 4 prozora:
 - **Project Explorer** prozor prikazuje strukturu svih trenutno otvorenih dokumenata. Svaki otvoren dokument predstavlja jedan *projekat*.
 - **Code** prozor sadrži VBA programe. Svaka stavka iz Project Explorer prozora ima pridružen kodni prozor.
 - **Immediate** prozor služi za direktno izvršavanje naredbi (VBA naredbi, matematičkih izraza, poziva funkcija).
 - **Properties** prozor prikazuje osobine trenutno izabranog objekta u Project Explorer prozoru.
- Prikaz prozora pomoću stavke **View** iz linije menija VBE-a ili preko odgovarajućih prečica.

Procedure

- VBA programski kôd se unosi u formi procedure.
- Procedure se dijele na **subprocedure** (makroi) i **funkcije**.
- Subprocedure predstavljaju skup VBA naredbi kojima se izvršava određeni zadatak.
 - Subprocedure imaju uticaj na svoje okruženje (npr. mogu mijenjati format ćelija u Excel-u).
 - Subprocedure **ne vraćaju rezultat!**
- Funkcije izvršavaju određeni zadatak i pritom vraćaju rezultat.
- Prvo ćemo detaljno objasniti funkcije, a nakon toga subprocedure.

Funkcije

- Funkcije primaju argumente, odrade određenu operaciju sa njima i vrate rezultat.
- Iako mogu uticati na izgled okruženja, **dobro definisana funkcija ne utiče na svoje okruženje!**
- Format funkcija je:

```
Function ImeFunkcije(arg1 As Tip, arg2 As Tip, ...) As Tip  
    VBA naredbe  
    ImeFunkcije = VraćenaVrijednost  
End Function
```

- Funkcije se najčešće definišu u okviru standardnog kodnog modula.
- Standardni kodni modul se u projekat ubacuje sa **Insert / Module**.

Primjer funkcije

- Dajmo primjer funkcije za sabiranje kvadrata dva unesena broja.

```
Function ZbirKvadrata(A As Integer, B As Integer) As Integer
    Dim C As Integer
    C = A ^ 2 + B ^ 2
    ZbirKvadrata = C
End Function
```

- **Ime funkcije predstavlja izlaznu promjenljivu**, tj. vrijednost koju vraća funkcija.
- Šta bi se desilo ako bi se izostavila posljednja linija u prethodnoj funkciji?

Poziv funkcije

- Funkcija definisana u standardnom kodnom modulu je dostupna svim procedurama u projektu.
- Funkcija se poziva navođenjem imena funkcije sa listom ulaznih argumenata u zagradi, razdvojenih zarezima.
- U prethodnom primjeru, poziv bi mogao biti `ZbirKvadrata(4,6)`.
- Ako se funkcija poziva iz Immediate prozora, mora se navesti naredba `Print` ispred poziva funkcije. Na primjer:
`Print ZbirKvadrata(4,6)`
- Umesto naredbe `Print`, može se koristiti njen skraćeni oblik `?`, tj.
`? ZbirKvadrata(4,6)`
- Pozivanje funkcije unutar ćelije radnog lista u Excel-u je isto kao pozivanje ugrađenih funkcija, tj. prvo znak `=`, pa ime funkcije sa argumentima.

Imenovanje funkcija

- Prilikom davanja imena funkciji moramo ispoštovati par jednostavnih pravila:
 1. Ime funkcije ne može biti duže od **255** karaktera,
 2. Prvi karakter mora biti slovo, a ostali karakteri mogu biti slova, cifre ili karakter podvlaka (_). Ostali karakteri nisu dozvoljeni.
- Na primer, dozvoljena imena funkcija su:
Izraz, Para5, Debeli_lad
dok su nedozvoljena imena:
5Para, Mirko&Slavko, Debeli lad
- Što se tiče imena funkcija, **VBA nije case-sensitive**, tj. ne pravi razliku između velikih i malih slova. Tako se imena **fun, Fun i FUN** odnose na istu funkciju.

Argumenti funkcije

- Ulazni argumenti su podaci koje se prosljeđuju funkciji prilikom poziva funkcije.
- Preko ulaznih argumenata, okolina (korisnik ili program) komunicira sa funkcijom.
- Na primjer, ako želimo da odredimo sinus broja X , potrebno je da pozovemo funkciju `Sin` kao `Sin(X)`, gde X predstavlja ulazni argument.
- Argumenti se navode u malim zagradama, i razdvajaju se zarezima:

`Function ImeFunkcije(arg1 As Tip, arg2 As Tip, ...) As Tip`

- Ulazne argumente definiše **ime** i **tip**.
 - Ime argumenta podliježe istim ograničenjima kao ime funkcije.
 - Tipovi predstavljaju tipove podataka koji odgovaraju argumentima. Tip se može izostaviti.

Tipovi VBA promjenljivih

- **Boolean**

Boolean promenljiva može imati dvije moguće vrijednosti, **True** (logički tačno) ili **False** (logički netačno). Zauzima 2 bajta.

- **Integer**

Cijeli brojevi iz opsega od -32 768 do 32 767. Zauzima 2 bajta.

- **Long**

Cijeli brojevi iz opsega od -2 147 483 648 do 2 147 483 647. Zauzima 4 bajta.

- **Single**

Brojevi u pokretnom zarezu (IEEE format) iz opsega -3.402823×10^{38} do 3.402823×10^{38} . Zauzima 4 bajta.

- **Double**

Brojevi u pokretnom zarezu iz opsega $4.94065645841247 \times 10^{-324}$ do $1.79769313486231 \times 10^{308}$. Zauzima 8 bajta.

Tipovi VBA promjenljivih (nastavak)

- **String**

Tip promjenljive za smještanje teksta.

- **Byte**

Koristi se za opseg cijelih brojeva od 0 do 255. Zauzima 1 bajt.

- **Date**

Koristi se za smještanje datuma i vremena. Zauzima 8 bajtova.

- **Currency**

Koristi se za rad sa novcem (velika preciznost). Zauzima 8 bajtova.

- **Object**

Koristi se za smještanje adresa objekata. Zauzima 4 bajta.

- **Variant**

Ovaj tip podataka se dodeljuje svim promenljivim čiji tip nije eksplicitno deklarisan. Promenljive tipa **Variant** mogu da sadrže većinu ostalih tipova podataka i tip se mijenja u zavisnosti od operacija.

Deklaracija VBA promjenljivih

```
Function ZbirKvadrata(A As Integer, B As Integer) As Integer
    Dim C As Integer
    C = A ^ 2 + B ^ 2
    ZbirKvadrata = C
End Function
```

- Unutar funkcije se mogu (ne moraju!) deklarirati promenljive.
- Deklaracija se (najčešće) vrši koristeći ključnu riječ **Dim**:
Dim Prom As Tip
- Ime promenljive **Prom** mora da zadovolji iste uslove kao ime funkcije, dok je **Tip** jedan od prethodno opisanih tipova.
- Iako se deklaracija obično vrši na početku funkcije (procedure), može se vršiti bilo gdje u kôdu.

Deklaracija VBA promjenljivih (nastavak)

- Pri deklaraciji više promjenljivih u jednom redu, potrebno je navesti tip za svaku promjenljivu pojedinačno:

`Dim X As Integer, Y As Double, S as String`

- Pri deklaraciji promjenljivih istog tipa, tip se mora navesti za svaku promjenljivu, npr.

`Dim X As Integer, Y As Integer`

Skraćeni oblik:

`Dim X, Y As Integer`

će deklarirati `Variant` promenljivu `X` i `Integer` promenljivu `Y`.

- Deklaracija se može forsirati opcijom **Require Variable Declaration** tabu **Editor** prozora **Tools / Options** u VBE.

Operator dodjele vrijednosti

- Operator dodjele vrijednosti je =
- Izraz koji se nalazi sa desne strane ovog operatora se izvrši i rezultat se dodjeljuje promjenljivoj ili objektu sa lijeve strane operatora. Na primjer:

Dim X As Double, Y As Double

X = 2.3

Y = (X - 1) ^ 2 - 3

- Izraz tipa

2.3 = X

nije dozvoljen jer se u konstantu ne može upisati vrijednost.

VBA aritmetički operatori

Operator	Operacija	Primjer	Rezultat
+	Sabiranje	10+4	14
-	Negacija	-10	-10
-	Oduzimanje	10-5	5
*	Množenje	10*7	70
/	Dijeljenje	13/5	2.6
\	Cjelobrojno dijeljenje	13\5	2
^	Stepenovanje	10 ^ 3	1000
Mod	Modulo	13 Mod 5	3

VBA operatori poređenja

Operator	Operacija	Primjer	Rezultat
=	Jednako	7 = 3	False
<>	Različito	7 <> 3	True
>	Veće od	7 > 3	True
>=	Veće od ili jednako	"a" >= "b"	False
<	Manje od	"aa" < "ab"	True
<=	Manje od ili jednako	"a" <= "b"	True
Like	Like	"VBA" Like "V?A"	True

- **OPREZI!** Operator provjere jednakosti je isti kao operator dodjele vrijednosti.

VBA logički operatori

Operator	Operacija	Primjer	Rezultat
And	I	Izr1 And Izr2	True ako su Izr1 i Izr2 True; inače False.
Or	ILI	Izr1 Or Izr2	True ako je Izr1 ili Izr2 True; inače False.
Xor	Ekskluzivno ILI	Izr1 Xor Izr2	False ako su Izr1 i Izr2 oba True ili oba False; inače True.
Not	Negacija	Not Izr	False ako je Izr jednak True i obrnuto.

Prvenstvo operatora

Operator	Rezultat
^	Stepenovanje
-	Negacija
* i /	Množenje i dijeljenje
\	Cjelobrojno dijeljenje
Mod	Modulo
+ i -	Sabiranje i oduzimanje
&	Nadovezivanje stringova
= <> <= >= <> Like Is	Operacije poređenja
And Or Xor Not	Logičke operacije

Matematičke funkcije

Funkcija	Vraćena vrednost
Abs(X)	Apsolutna vrijednost broja X.
Sin(X)	Sinus broja X.
Cos(X)	Kosinus broja X.
Tan(X)	Tangens broja X.
Atn(X)	Arkus tangens broja X.
Exp(X)	Eksponent broja X, tj. e^X .
Log(X)	Prirodni logaritam broja X.
Sqr(X)	Kvadratni korijen broja X.
Sgn(X)	Znak broja. Vraćena vrednost je 1 ako je $X > 0$, -1 ako je $X < 0$ i 0 ako je $X = 0$.
Fix(X)	Cijeli dio broja X. Ako je $X < 0$, Fix vraća prvi negativni broj veći od ili jednak X.
Int(X)	Cijeli dio broja X. Ako je $X < 0$, Int vraća prvi negativni broj manji od ili jednak X.
Rnd()	Slučajan broj između 0 i 1.

Forsirani izlazak iz funkcije

- Izvršavanje funkcije se može prekinuti sa **Exit Function**.
- U slučaju prekida izvršavanja, voditi računa da je imenu funkcije, kao izlaznoj promjenljivoj, dodijeljena vrijednost prije **Exit Function**.

```
Function Fun(Lista argumenata) As Tip  
    VBA naredbe  
    Fun = VraćenaVrijednost  
    Exit Function  
    VBA naredbe  
End Function
```

- Alternativno, možemo koristiti naredbu **End** umesto **Exit Function** za izlazak iz funkcije.

VBA komentari

- Komentar u VBA počinje apostrofom (').

' Ovo je komentar

' Ovo je komentar

X = 2.3 ' Ovo je takođe komentar

- Za komentar se rjeđe koristi ključna reč **Rem**.

Rem Ovo je komentar

X = 2.3 : Rem Ovo je takođe komentar

Ukoliko se **Rem** komentar nalazi u nastavku naredbe, naredba se mora završiti dvotačkom (:).

Više naredbi u jednom redu.

Prelamanje naredbe

- Za grupisanje više naredbi u jednom redu, koristi se dvotačka (:).

$X = 2.3 : Y = X^2 : Z = X - Y$

- Za prelamanje naredbe u više redova, koristi se podvlaka (_).

$X = X + 2.3$ _
 $+ Y^3$

Primjer funkcije

- Napisati funkciju **Izraz** koja za ulazni argument ima realan broj dvostruke preciznosti X i vraća vrijednost izraza:

$$Y = \frac{\sqrt{X^4 + 3}}{X^3 - 4} - e^{|\sin(X^2)|}$$

```
Function Izraz(X As Double) As Double
```

```
    ' Racunanje izraza
```

```
    Dim Y as Double
```

```
    Y = Sqr(X ^ 4 + 3) / (X ^ 3 - 4) - Exp(Abs(Sin(X ^ 2)))
```

```
    Izraz = Y
```

```
End Function
```

Makro rekorder

- Pomoću makro rekordera kreirati makro sa imenom **PrviMakro** koji u ćeliju C3 upisuje tekst VBA i toj ćeliji mijenja font na Courier New, veličine 11 pt. Editovati makro **PrviMakro**.
- Mane makro rekordera:
 1. Kôd koji generiše makro rekorder ima puno suvišnih naredbi.
 2. Sam kôd zavisi od određenih podešavanja, tj. ista radnja se može kodirati na više načina.
 3. Makro rekorder ne može generisati dodelu vrijednosti promjenljivoj, programsku petlju (ponavljanje naredbi), uslovno izvršavanje, prikaz dijalog prozora itd.
 4. Ne može se kreirati funkcija.
- Makro rekorder je pogodan alat za snimanje jednostavnih makroa ili malih dijelova složenijih makroa.