

## VBA

### 3. vježba

#### For-petlja

U slučajevima kada je neki postupak potrebno ponoviti više puta sa različitim parametrima, koriste se petlje. Primjerice, za zbrajanje brojeva od 1 do 100 to se radi tako da se krene od 0 i u svakom koraku dodaje se sljedeći broj. Riječima to izgleda ovako:

*suma = 0*

*za svaki broj od 1 do 100: sumi dodaj taj broj, a zatim idi na sljedeći broj*

U programskim jezicima za takve situacije najčešće se koristi For-petlja. Njena sintaksa u VBA je:

```
For brojac = pocetak To kraj
    naredbe
Next brojac
```

gdje je FOR početak petlje, brojac je varijabla koja mijenja svoju vrijednost svakim prolaskom kroz petlju, pocetak je početna vrijednost brojača (najčešće 1), kraj je posljednja vrijednost brojača, nakon koje se izvode sljedeće naredbe koje su napisane nakon NEXT naredbe. Naredba unutar petlje izvršava neku radnju u ovisnosti o trenutnoj vrijednosti brojača. Naredba NEXT vraća program na FOR s tim da brojač postane za jedan veći. Sumiranje brojeva od 1 do 100 pomoću for-petlje stoga izgleda ovako:

```
suma = 0
For i = 1 To 100
    suma = suma + i
Next i
```

Naredbeni redak `suma = 0` naziva se INICIJALIZACIJA SUME. To je definiranje prvog sumanda i činjenica da iznosi 0 ne mijenja konačni rezultat ( $0 + n + m + \dots = n + m + \dots$ )  
Općenito,

$$\sum_{i=a}^b f$$

gdje je  $f$  neki izraz ovisan o brojaču  $i$  računa se petljom:

```
suma = 0
For i = a To b
    suma = suma + f
Next i
```

Općenito,

$$\prod_{i=a}^b f$$

gdje je  $f$  neki izraz ovisan o brojaču  $i$  računa se petljom:

```
produkt = 1
For i = a To b
    produkt = produkt * f
Next i
```

Ako je potrebno for-petlju izvoditi tako da se u svakom koraku brojač poveća za iznos  $k$ , sintaksa je:

```
For brojac = pocetak To kraj Step k
    naredbe
Next brojac
```

(negativna vrijednost  $k$  uzrokuje brojanje unatrag!)

## Range

U 2. vježbi pokazano je da se vrijednosti zapisanoj u ćeliji A1 u Sheet1 pristupa preko naredbe `Worksheets("Sheet1").Range("A1").Value`. Ukoliko je potrebno istovremeno pristupiti nekom rasponu ćelija, primjerice da se u više ćelija upiše ista vrijednost, taj raspon na standardni način navodi se u Range-u. Tako primjerice naredba

```
Worksheets("Sheet1").Range("A1:D25").Value = "Tikvica"
```

ispisuje riječ *Tikvica* u ćelije od A1 (gornji lijevi kut) do D25 (donji desni kut).

## Activate

Obzirom da je često naporno pisati `Worksheets("Sheet1")`, ako je poznato da će program cijelo vrijeme izvršavati naredbe samo unutar Sheet1, može se na početku programa navesti naredba:

```
Worksheets("Sheet1").Activate
```

Nakon te naredbe više se ne mora u adresiranju navoditi `Worksheets`, te se za pristup npr. ćeliji A1 u Sheet-u 1 može umjesto `Worksheets("Sheet1").Range("A1").Value` jednostavno pisati `Range("A1").Value`. Upotreba `Worksheets...` opcije unutar FOR-NEXT uzrokuje grešku u izvođenju programa, stoga u tom slučaju treba koristiti `Activate`, a da bi naredba `Activate` funkcionirala, program mora biti napisan u MODULU

## Cells

Ponekad je potrebno da program ispisuje vrijednosti u ćelije ovisno o nekim parametrima, primjerice: ispisati jedinice na pozicije kojima je broj retka jednak broju stupca, a inače nule. U takvim slučajevima ne mogu se unaprijed znati na koji Range se naredba odnosi, pa se raspon za Range opisuje putem svojstva Cells.

Tako `Range(Cells(a,b),Cells(c,d))` opisuje raspon ćelija od kojih je gornja lijeva u retku a i stupcu b, a donja desna u retku c i stupcu d.

**Primjer.** Program koji u prvi stupac (Sheet2) ispisuje kvadrate prvih  $n$  prirodnih brojeva, pri čemu broj  $n$  zadaje korisnik putem InputBox-a.

```
Option Explicit
Sub kvadrati()
Dim n As Integer, i As Integer
Worksheets("Sheet2").Activate
n = InputBox("Unesite neki broj")
For i = 1 To n
Range(Cells(i,1),Cells(i,1)).Value = i^2
Next i
End Sub
```

---

## Zadaci uz vježbu 2:

- 1) Napišite program koji računa  $n!$  ( $n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$ ). Broj  $n$  treba unijeti korisnik putem InputBox-a. Ispis rezultata putem Message box-a.
- 2) Napišite program koji računa sumu prvih  $n$  kubova prirodnih brojeva. Broj  $n$  treba učitati iz ćelije A1 (Sheet1), a traženu sumu treba ispisati u ćeliju A3 (Sheet1).
- 3) Napišite program koji broj kojeg unese korisnik (putem InputBox-a) ako je veći od 1 unosi u ćelije od A1 do A10, a inače u ćelije od B1 do H1.
- 4) Napišite program koji u prvi redak u Sheet1 ispisuje kvadrate parnih brojeva od 2 do 50. (Hint: koristite jedan FOR za definiranje članova i za adresiranje uz STEP za preskakanje brojeva. Indeks stupca možete podijeliti s dva da se stupci ne preskaču: `Range(Cells(1,n/2),Cells(1,n/2)).Value = n`).
- 5) Napišite program koji koji na dijagonalne pozicije u Sheet2 ispisuje recipročne brojeve od 1 do  $n$ . Broj  $n$  unosi korisnik putem InputBox-a. (Hint: dijagonalne pozicije imaju adresu u kojima su indeks retka i stupca jednaki).