Објектно орјентисано програмирање

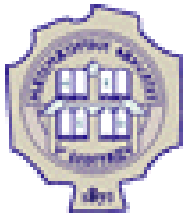
Владимир Филиповић
vladaf@matf.bg.ac.rs
Александар Картељ
kartelj@matf.bg.ac.rs

Низови у програмском језику Јава

Владимир Филиповић
vladaf@matf.bg.ac.rs
Александар Картељ
kartelj@matf.bg.ac.rs

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

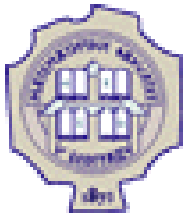


Низови у Јави

- Низ се дефинише као група променљивих истог типа које се појављују под заједничким именом.
- Низовни тип података у Јави има следећа својства:
  1. садржи линеарно уређен, унапред познат, број чланова,
  2. сви чланови су истог типа и имају заједничко име; чланови могу бити примитивног или објектног типа,
  3. сваком члану приступа се помоћу заједничког имена низа и индекса члана,
  4. сви индекси су целобројног типа,
  5. сви чланови низа се третирају као посебне променљиве (називају се и индексним променљивим).

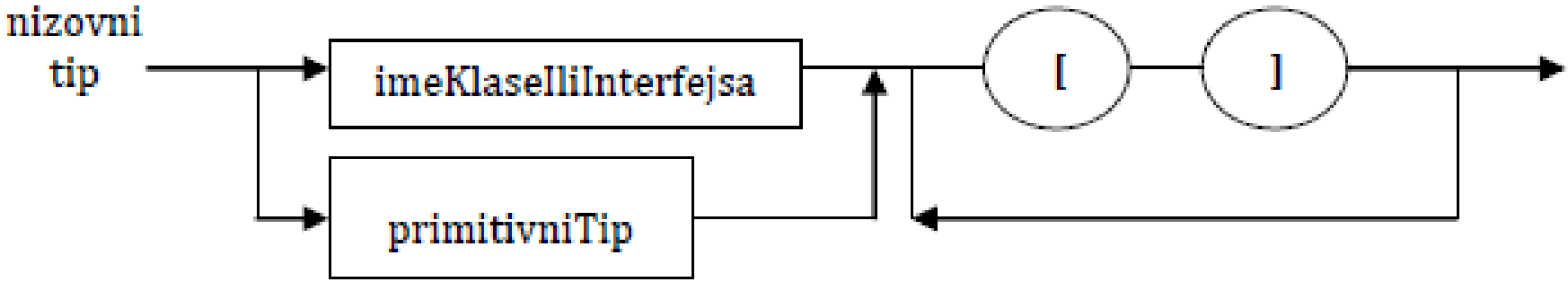
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Низови у Јави (2)

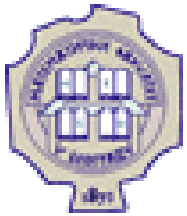
- Синтакса низовног типа се дефинише на следећи начин:



- Низовни тип у Јави је увек објектни тип, тј. низ је увек објекат.

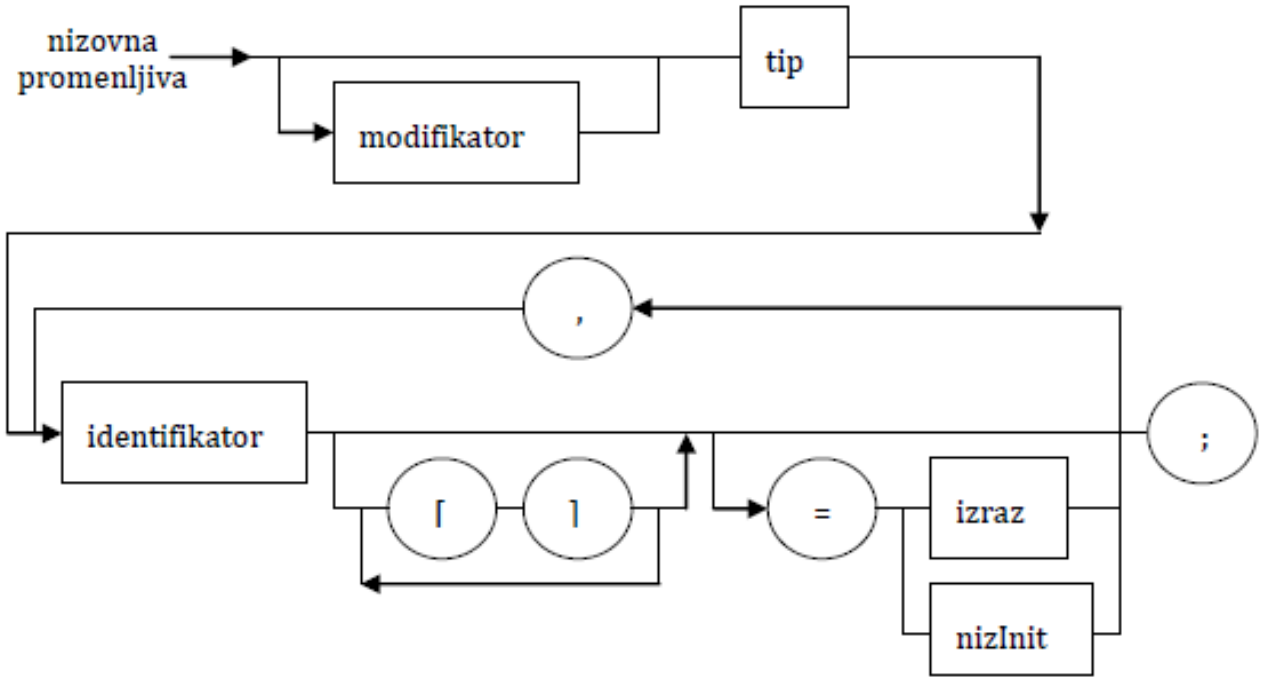
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



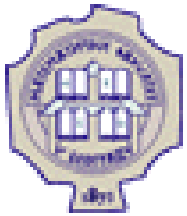
Низови у Јави (3)

- Низовна променљива се декларише у делу програма за декларацију.
- При томе се могу задати и вредности члановима низа.



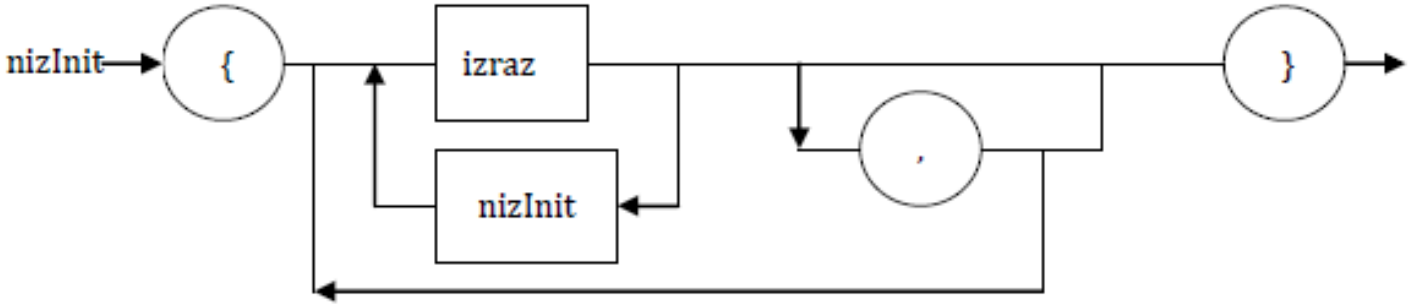
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

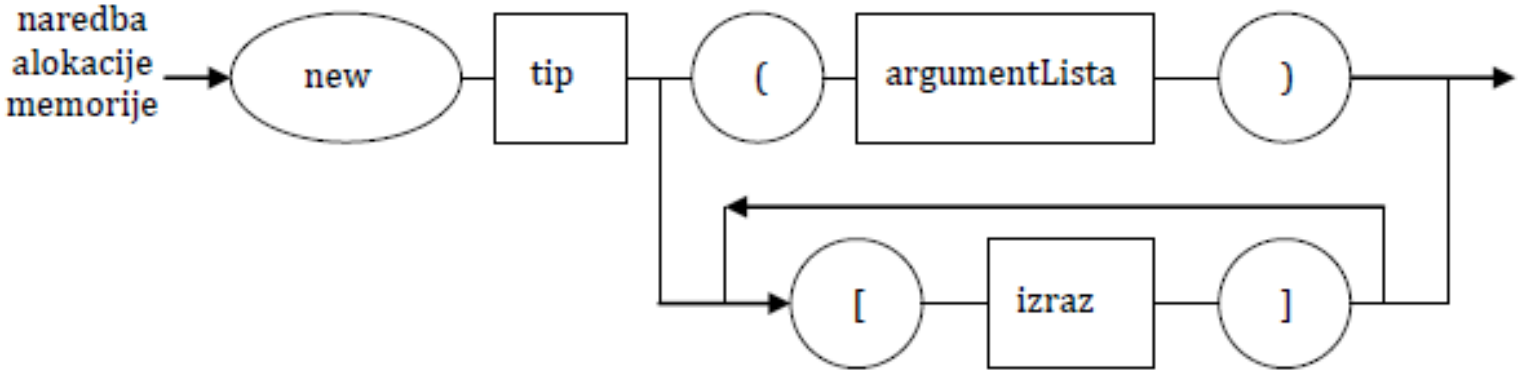


Низови у Јави (4)

- Додељивање почетних вредности (иницијализација) члановима низа врши се у делу декларације названом nizInit.

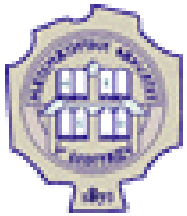


- Синтакса наредбе за алокацију меморије је следећа:



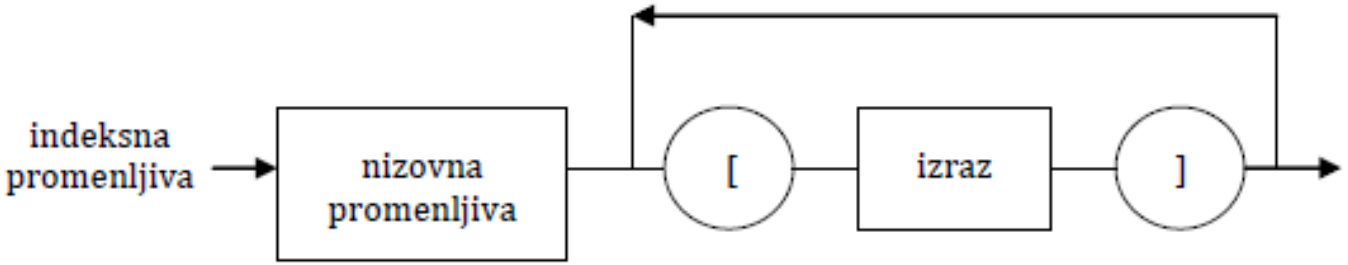
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

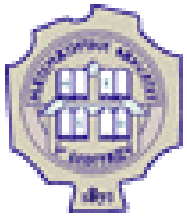


Низови у Јави (5)

- Елементима низа (индексним променљивима) приступа се помоћу индекса.

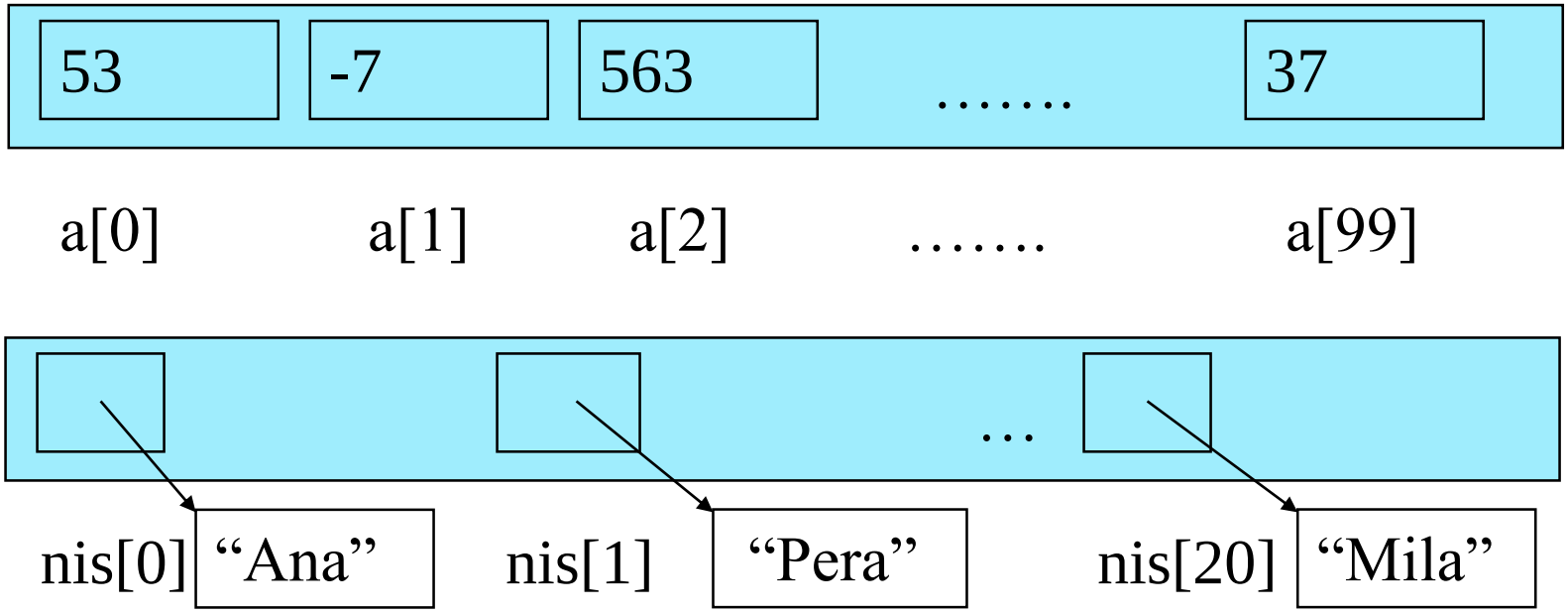


- Са индексним променљивима оперише се на исти начин као и са променљивима без индекса.
- Дакле, вредности индексних променљивих се могу мењати помоћу наредбе додељивања и користити у изразима.



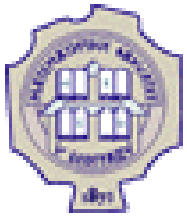
Једнодимензионални низ

- Једнодимензионални за приступ елементима користи тачно један индекс.



Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Једнодимензионални низ (2)

Декларисање низова (постоје два начина):

Пример.

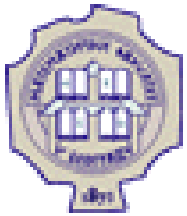
```
int pozicija[];
String niska[];
Knjiga naslov[];

// Iste deklaracije na drugi način
int [] pozicija;
String [] niska;
Knjiga [] naslov;
```

Уочава се разлика између ове две врсте декларација:

```
int [] a, b, zzz; // sve 3 promenljive a, b i zzz se deklarisu kao
                  //nizovne
int a, b, zzz[]; // samo zzz se deklarise kao nizovna promenljiva
```

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Једнодимензионални низ (3)

Креирање низова

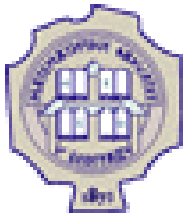
- Постоје два начина:

  (1) помоћу оператора new,

  (2) набрајањем чланова

Пример.

```
Point [] tacke = new Point[20]; // prvi nacin
int [] brojac = new int[100]; // prvi nacin

// Sledi drugi nacin kreiranja
String godDoba[], s="mika";
godDoba ={"prolece", s, "leto", "jesen", "zima"};
int celi[] = { 23, 46, -123, 17, 36, -12};
```



Једнодимензионални низ (4)

Приступ члановима низа

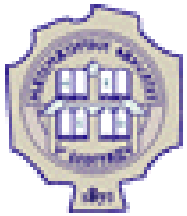
- Нумерисање чланова низа увек почиње од 0.
- Члановима низа се приступа помоћу: *ImeNiza[indeks]*
- *ImeNiza* – име које се појављује у декларацији
- *indeks* – целобројни литерал или израз

Пример.

```
int [] brojac = new int[100];
String [] godDoba={“prolece”, “leto”, “jesen”, “zima”};
…
brojac[50]=2; // Ne bi valjalo: brojac[100] = 3;
a=brojac[i+j];
x = godDoba[3];
```

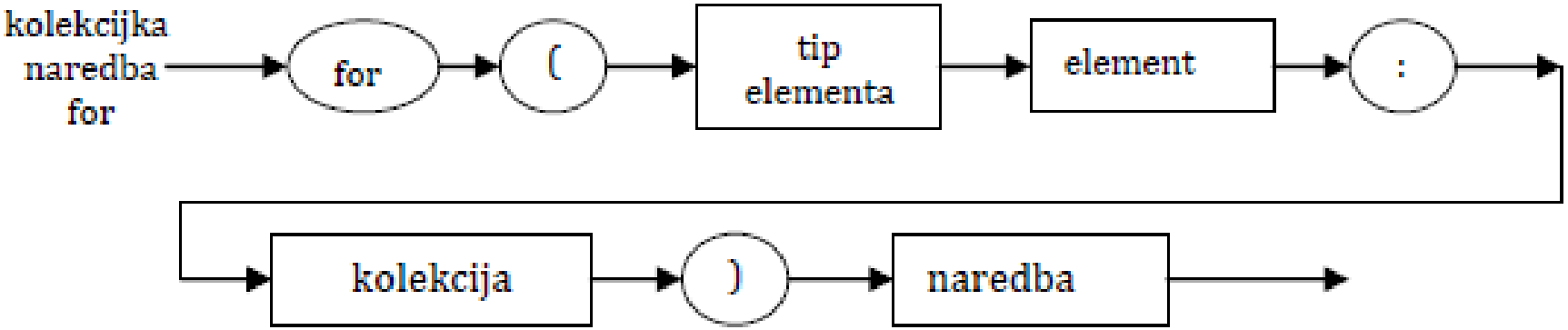
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



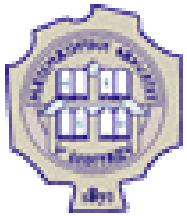
Колекцијска наредба for

- Колекцијска наредба for служи за пролазак кроз колекцију/низ.
- Притом променљива у наредби for не представља бројач, већ редом узима вредности елемената низа тј. колекције.
- Колекцијска наредба for има следећу синтаксу:



Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Колекцијска наредба for (2)

Овде важи следеће:

```
<tip elementa>  ::= <tip>
<elem>          ::= <ime>
<kolekcija> ::= <izraz>
```

Пример.

```
{ ...
        int n = sc.nextInt();
        double a[] = new double[n];
        …
        double najveci=a[0];
        for (double elem: a)
                if ( elem > najveci )
                        najveci = elem;
        ...
}
```

Аргументи командне линије

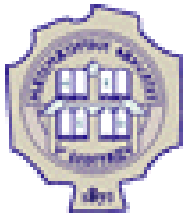
- Аргументи комадне линије омогућавају да се приликом покретања јава програма, том програму проследе параметри.
- Улазна тачка програма, тј. статички метод main има следећу синтаксу:

```
public static void main(String[] argumentiKomandneLinije) {
        // naredbe
}
```

- Наредбе и телу програма могу референсати на једнодимензионални низ ниски argumentiKomandneLinije који ће садржавати аргументе специфициране у комадној линији
  - елеменат низа са индексом 0 представља први прослеђени аргумент, са индексом 1 други итд.
  - приликом парсирања комадне линије, размак тј. ' ' раздваја, а ако је потребно да параметар садржи ' ', користи се '"'

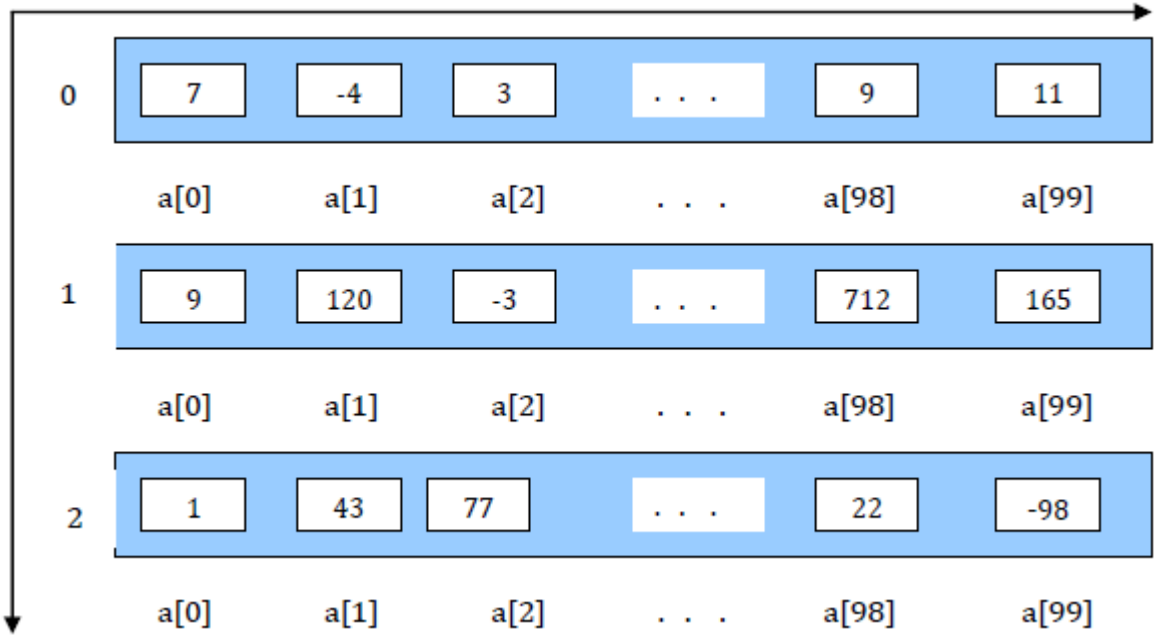
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



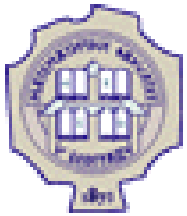
Дводимензионални низ

- Дводимензионални низ чине елементи истог типа и сваком од њих приступа се помоћу имена низа и два индекса.
- Више једнодимензионалних низова се могу поређати један испод другог као на следећој слици:



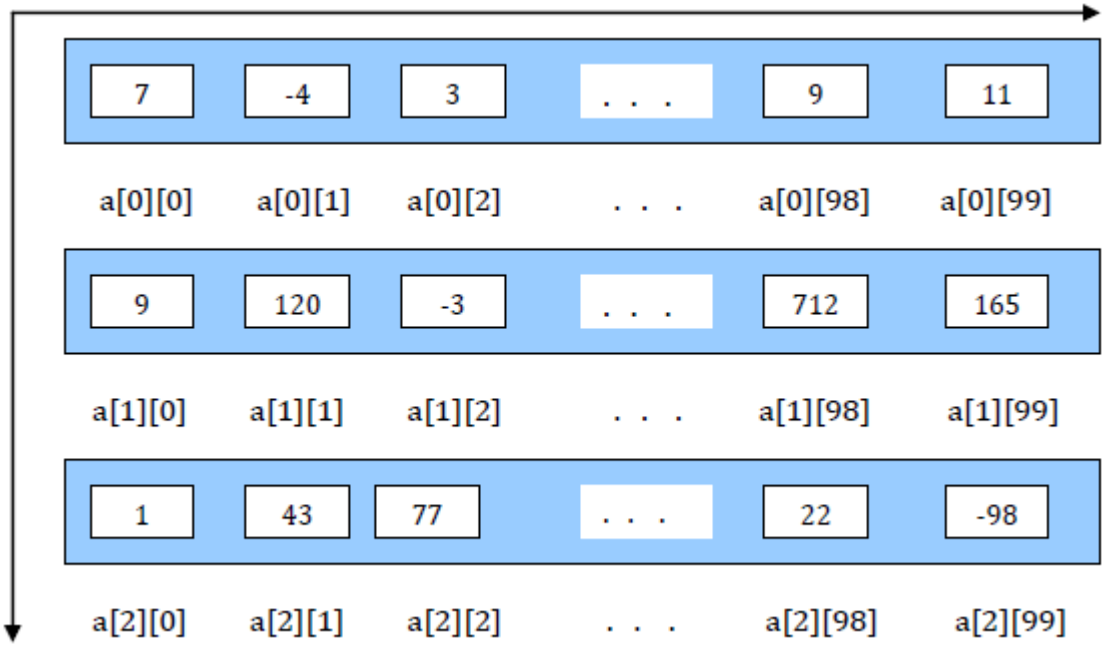
Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Дводимензионални низ (2)

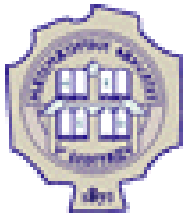
- Индексирање употребом два индекса.



- Дводимензионални низ представља једнодимензионални низ чији елементи су једнодимензионални низови.

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Дводимензионални низ (3)

- Декларацију имена низа можемо да извршимо на два начина:

```
int m, a[][]; // prvi nacin
int[][] c, b; //drugi nacin
```

- Након декларације низовне променљиве, креирамо низовни објекат, тј. вршимо резервисање меморијског простора за смештање чланова низа:

```
a = new int[50][100];
```

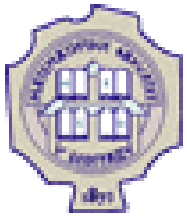
Овом наредбом је резервисано 5000 целобројних локација у меморији.

- Декларацију низовне променљиве и резервисање меморијског простора могли смо извршити једном наредбом, тј. могли смо писати:

```
int a[ ][ ] = new int[50][100];
```

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs



Вишедимензионални низ

- Дводимензионални низови представљају један, најчешће коришћен, случај вишедимензионалних низова.
- За низове чији је број димензија већи од два, принцип рада је потпуно исти као и са дводимензионалним низовима.
- Дакле, као што су формирани дводимензионални низови, на исти начин се могу формирати тродимензионални, четвородимензионални итд.
- Већ смо уочили да оперисање са матрицама можемо свести на оперисање са једнодимензионалним низовима, то важи и за све вишедимензионалне низове.

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

Коришћење неких метода класе Arrays

- Класа java.util.Arrays садржи различите методе за манипулацију са низовима:
  - Попуњавање низа – доделу специфициране вредности сваком члану низа – статички метод fill()
  - Уређење (сортирање) низа – премештање елемената у оквиру низа, тако да сви елементи буду уређени у нерастући поредак - статички метод sort()
  - Претрагу елемената у (сортираном) низу – методом бинарне претраге се одређује позиција датог елемента у сортираном низу - статички метод binarySearch()

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

Методи са променљивим бројем аргумената

- Новије верзије програмског језика Јава допуштају да параметри функције не буду строго фиксирани, већ да једна иста функција може бити позвана и са различитим бројем аргумената истог типа:
  - Тада је параметар функције низ, а приликом позива се узастопни аргументи истог типа аутоматски конверују у низ
  - Налог за аутоматску конверзију узастопних аргумената датог типа у низ се постиже декларисањем помоћу **...**
  - Овако дефинисан параметар функције мора да буде последњи у списку параметара.

Математички факултет

{vladaf, kartelj}@matf.bg.ac.rs

Захвалница

Велики део материјала који је укључен у ову презентацију је преузет из презентације коју је раније (у време када је он држао курс Објектно орјентисано програмирање) направио проф. др Душан Тошић.

Хвала проф. Тошићу што се сагласио са укључивањем тог материјала у садашњу презентацији, као и на помоћи коју ми је пружио током конципцирања и реализације курса.