

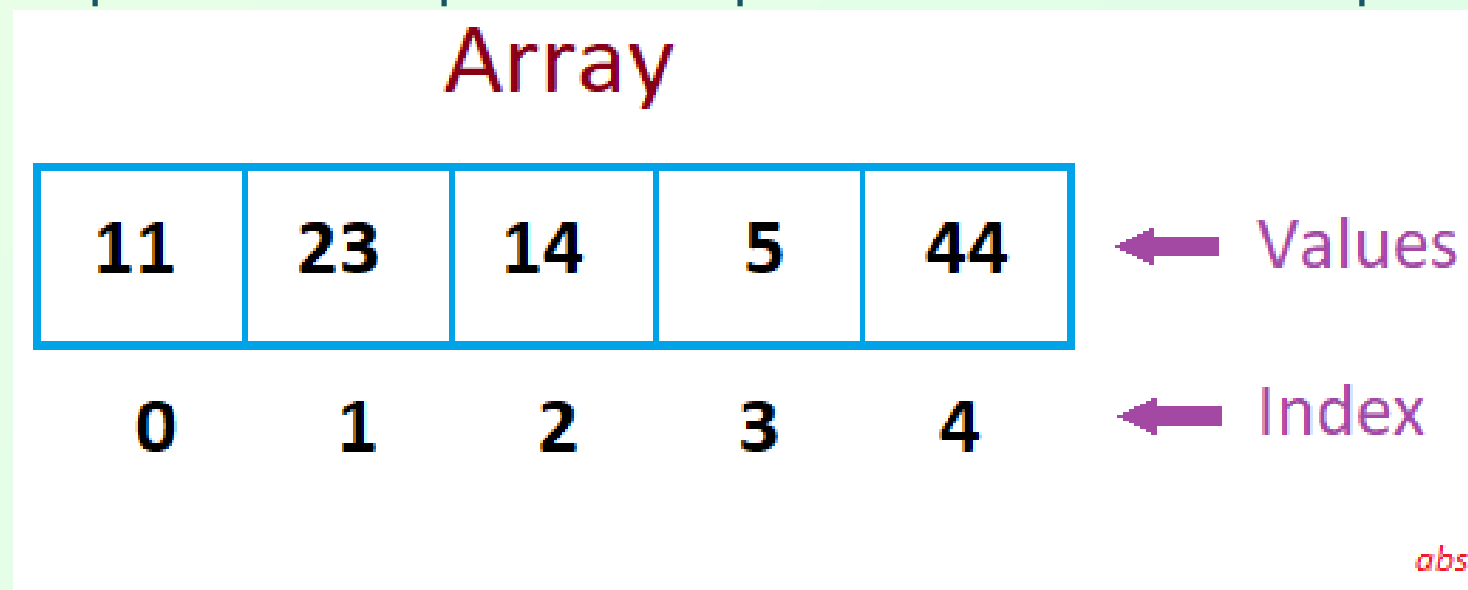
Nizovi, matrice i stringovi u Javi

Objektno orijentisano programiranje
Vežbe 2

Đurđa Milošević
Matematički fakultet

Nizovi

- Niz je struktura podataka koja čuva elemente istog tipa u uzastopnim memorijskim lokacijama
- Prednosti niza u odnosu na druge strukture podataka:
 - Random access - slučajni pristup
 - Efikasna iteracija kroz elemente
 - Cache friendliness - Dobra iskorišćenost memorije (lokalnost)
- Mane:
 - Zahtevaju kontinualan blok memorije
 - Operacije dodavanja i brisanja elemenata su skupe



Adresiranje- Kako se računa adresa elementa?

$$\text{addr} = a + i * \text{sizeof}(T)$$

Gde je:

- a – početna adresa niza
- i – indeks elementa (počinje od 0)
- T – tip elementa
- sizeof(T) – veličina jednog elementa u bajtovima

Nizovi u Javi

Deklaracija niza:

1. `int[] arr;` ← ovaj način se češće koristi
2. `int arr[];`

Inicijalizacija niza:

```
int[] arr = new int[size];
```

- `arr` je referenca na niz
- memorija se alocira pomoću `new` (na hipu)
- veličina niza je fiksna
- elementi imaju podrazumevane vrednosti (numerički tipovi - 0, boolean - false...)

Niz je moguće inicijalizovati i na sledeći način:

```
int[] arr = {1, 2, 3};
```

- Niz se inicijalizuje direktno vrednostima
- Nema potrebe za `new`
- Dužina niza = broj elemenata
- Java automatski kreira niz u pozadini

java.util.Arrays

Arrays.sort(a);

Arrays.binarySearch(a, key);

Arrays.equals(a, b);

Arrays.toString(a);

- Utility klasa koja sadrži veliki broj statičkih funkcija za rad sa nizovima
- Nalazi se u paketu java.util
- Ova klasa nam omogućava da lako sortiramo niz, pretražujemo ga, upoređujemo nizove ili ih pretvorimo u string za ispis

Matrice u Javi

- Matrica = niz nizova
- Svaki red je poseban niz
- `int[][] arr = new int[3][4];`
- `a[i]` → i-ti red (niz)
- `a[i][j]` → element u i-tom redu i j-toj koloni
- Redovi mogu imati različitu dužinu (jagged array)

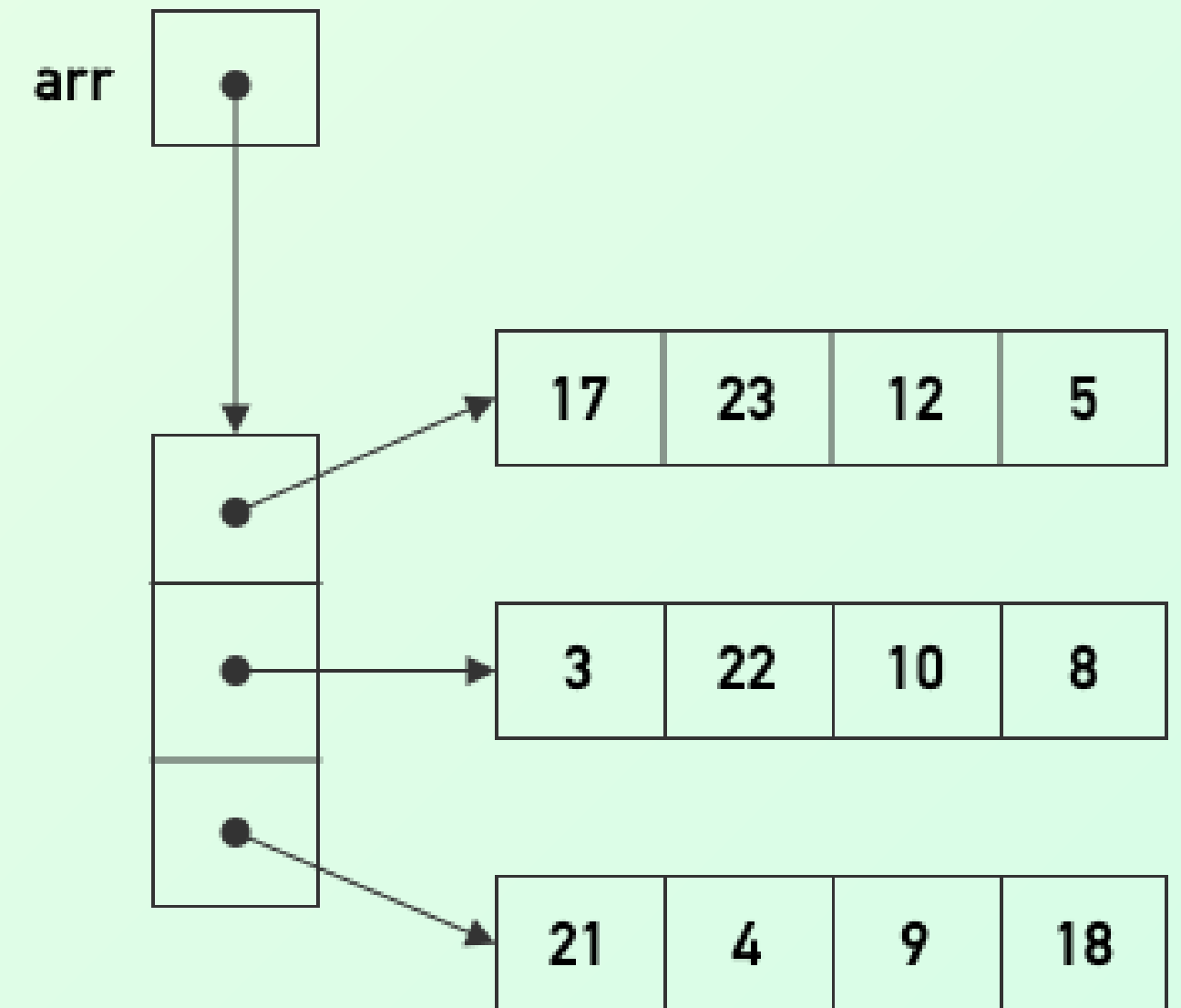
Šta označava `arr[0]`?

Šta označava `arr[1][1]`?

Šta označava `arr.length`?

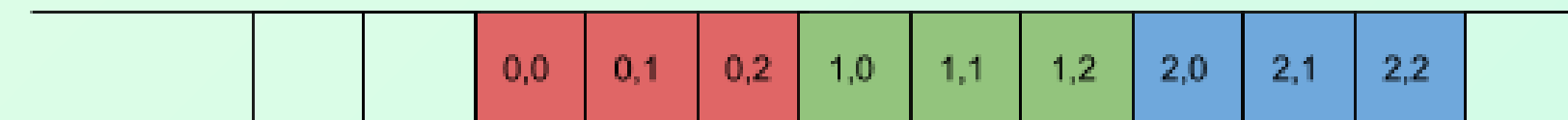
Šta označava `arr[0].length`?

Java:



C:

row,col	0,0	0,1	0,2
	1,0	1,1	1,2
	2,0	2,1	2,2



Stringovi u Javi

String:

- Klasni tip (java.lang.String)
- Predstavlja niz karaktera (UTF-16)
- Immutable (nepromenljiv)

s.length() // dužina

s.charAt(i) // karakter na poziciji i

s.equals(t) // poređenje sadržaja

s.substring(a,b) // podstring

s.toLowerCase() // mala slova

s.toUpperCase() // velika slova

Načini kreiranja:

```
String s1 = "Hello"; // literal (String pool)
```

```
String s2 = new String("Hello"); // heap
```

Razlika:

Literal (String pool)

→ koristi postojeći objekat ako postoji

→ optimizacija memorije

new (heap)

→ uvek pravi novi objekat

→ referenca pokazuje na taj objekat

String kao **niz karaktera:**

```
char[] niz = {'H', 'e', 'l', 'l', 'o'};
```

```
String s = new String(niz);
```

StringBuilder

- Klasni tip za rad sa promenljivim stringovima
- Za razliku od String → mutable (promenljiv)
- Koristi se kada često menjamo sadržaj

```
StringBuilder sb = new StringBuilder();
```

StringBuffer

- Sličan kao StringBuilder
- Thread-safe

Najvažnije metode:

```
sb.append(x) // dodavanje  
sb.insert(i, x) // ubacivanje  
sb.delete(a, b) // brisanje  
sb.reverse() // obrće string  
sb.toString() // konverzija u String
```

Hvala na pažnji