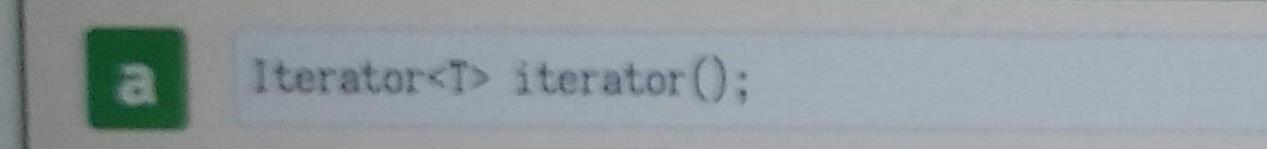
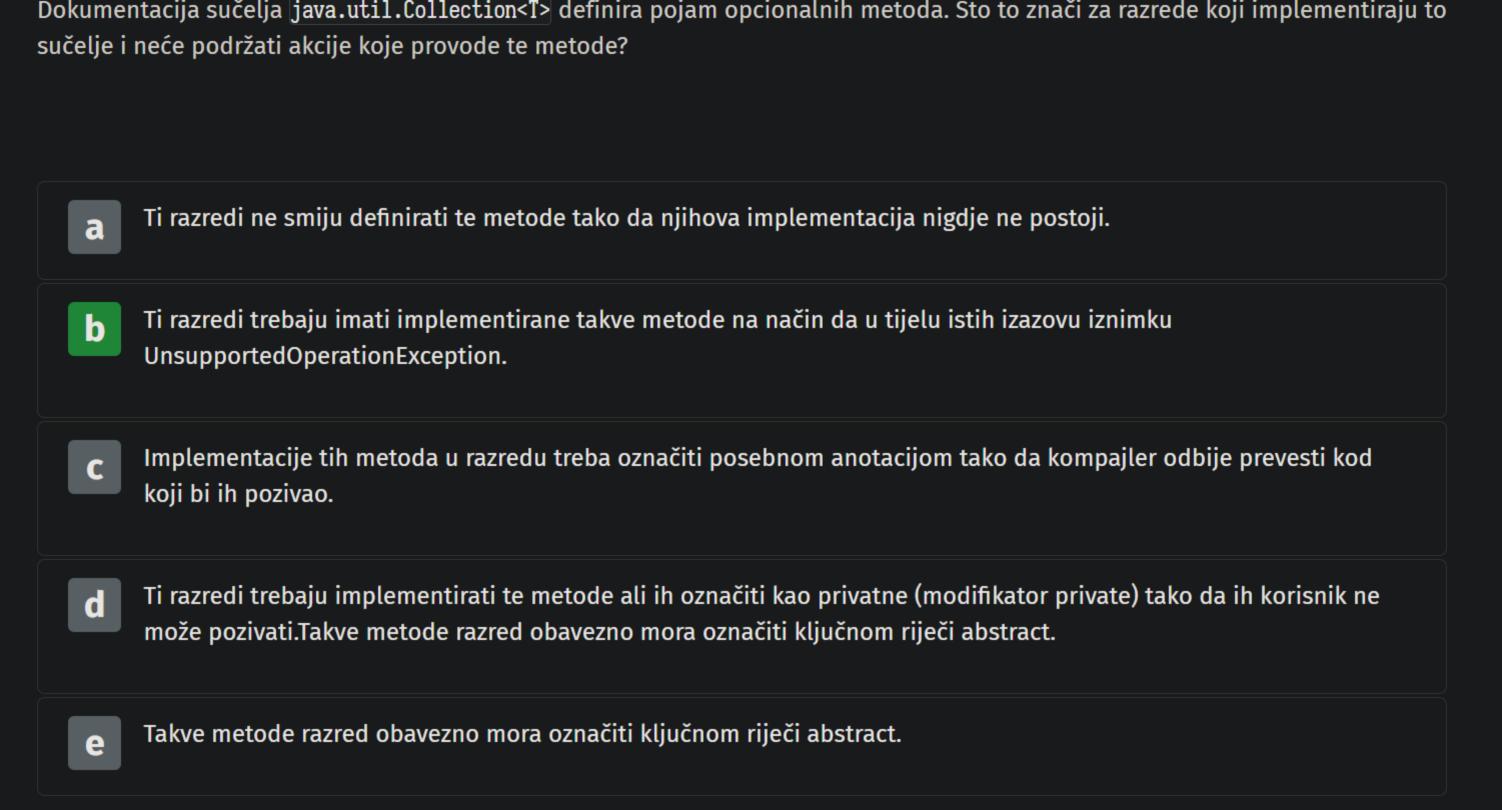
Koju apstraktnu metodu je potrebno nadjačati kod implementacije sučelja java.util.function.Predicate<T>?

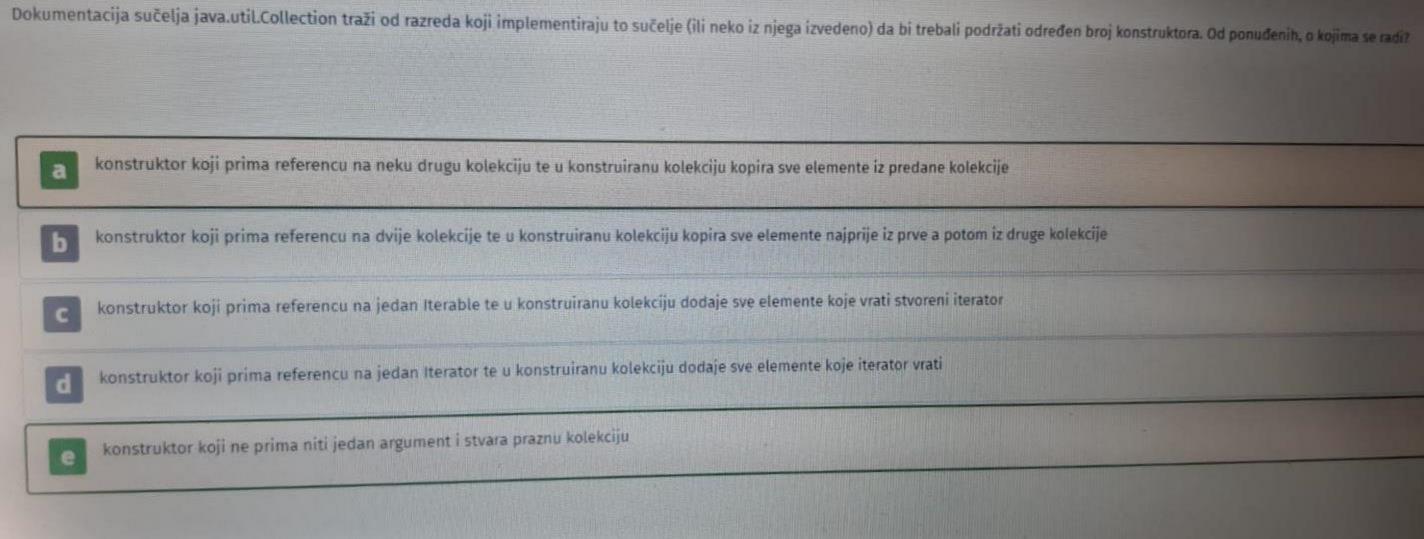
- boolean compare(T first, T second)
- boolean test(I t)
- Predicate(boolean b)
- void test(I t)
- int compareTo(I other)

	netode definimme suceljem java.lang.lterator<1+?
a	boolean hashext();
ь	Sterator <t> iterator();</t>
c	resure();
d	T mext();

Koje su metode definirane sučeljem java.lang.Iterable<T>?







Što od ponuđenoga vrijedi za sučelje java.util.Collection? Iterable<T> extends Collection<T> a Iterator<T> extends Collection<T> Collection<T> extends Iterable<T> Collection<T> extends Iterator<T>

Što od n	avedenog je istina za java.util.Collections?
a	java.util.Collections sadrži samo statičke metode
Ь	To je sučelje
c	To je statička metoda
d	To je metoda
е	To je klasa

Koja izjava vrijedi za java.util.LinkedList: lista elemente pohranjuje dinamički alocirajući nove čvorove za dvostruko povezanu listu Za I/O tokove u Javi vrijedi: Tokovi su jednosmjerni Klijent u izlazni tok može isključivo zapisivati Klijent iz ulaznog toka može isključivo čitati podatak po podatak, ne više njih Tokovi se mogu ulančati korištenjem dekoratora

Ispravan način definiranja toka podataka iz datoteke je:

```
tok = new FileInputStream(<ime_datoteke>)

tok = new InputStream(<ime_datoteke>)
```

tok = new OutputStream(<path\_do\_datoteke>)

```
tok = new File(<ime_datoteke>, Mode.Input)
```

Ispravan način definiranja toka podataka iz datoteke je:

a tok = new OutputStream(<path\_do\_datoteke>)

b tok = new InputStream(<path\_do\_datoteke>)

tok = Files.newInputStream(<path\_do\_datoteke>)

tok = new File(<path\_do\_datoteke>, Mode.Output)

Označite metode sučelja FileVisitor visitDirectory a postVisitDirectory b preVisitDirectory C visitFile d visitFileFailed e

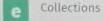
Znakovr	ni tokovi unutar paketa	a java.io n	nodelirani	su apstraktn	im razre	dima
a	Listeners					
b	Charset					
С	Writer					
d	Reader	1				



b Set



d Map



događja se to u jesenjoj naci kada pada kestenje po asfaltu i kada se cvju psi u daljini

Ako tiječi dodajamo u Lipkadki sebset radoslijedom kajim su poveda

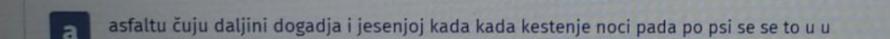
Ako riječi dodajemo u LinkedHashSet redoslijedom kojim su navedene u gornjoj rečenici, što čemo dobiti kad sadržaj kolekcije ispišemož

dogadja se to u jesenjoj noci kada pada kestenje po asfaltu i cuju psi daljini

- b asfaltu čuju daljini događja i jesenjoj kada kada kestenje noci pada po psi se se to u u
- asfaltu cuju daljini dogadja i jesenjoj kada kestenje noci pada po psi se to u
  - se u noci to jesenjoj kestenje po cuju pada psi dogadja asfaltu daljini i kada

dogadja se to u jesenjoj noci kada pada kestenje po asfaltu i kada se cuju psi u daljini

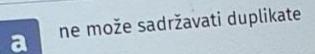
Ako riječi dodajemo u TreeSet redoslijedom kojim su navedene u gornjoj rečenici, što ćemo dobiti kad sadržaj kolekcije ispišemo?

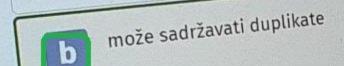


asfaltu cuju daljini dogadja i jesenjoj kada kestenje noci pada po psi se to u

Listu možem	no stvoriti pozivom:	
2 100	w ArrayListo();	
Ь	v List();	
C An	rays.asList("A", "B", "C");	
d Res	v LinkedList();	
e Arr	rays.asList(1,2,3);	

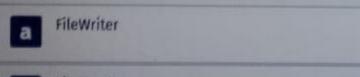
Lista je uređena kolekcija koja

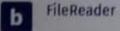


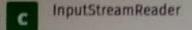


a	apstraktnim klasama InputStream i OutputStream
ь	sučeljima Input i Output
С	objektima System.in i System.out
d	apstraktnim klasama InputStream, OutputStream i ChannelStream

## Mostovi između znakovnih tokova i tokova okteta su



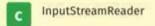




d OutputStreamWriter

Most između znakovnih tokova i tokova okteta je





d FileReader

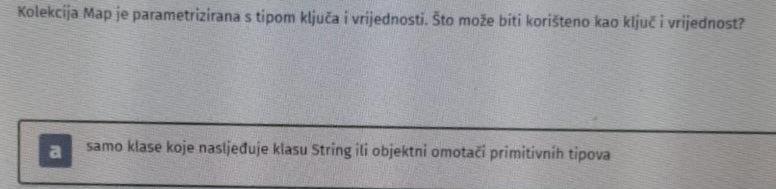
Koji od p	oonuđenih naziva predstavljaju klase koje implementiraju sučelje java.util.Set, a koje pripadaju Javinom okviru kolekcija?
a	HashSet
b	ArraySet
С	LinkedSet
d	TreeSet
You	

Koji od	ponuđenih naziva predstavljaju klase koj	e implementiraju sučelje java.u	itil.Set, a koje pripadaju Javino	m okviru kolekcija?
a	LinkedHashSet			
Ь	TreeSet			
c	HashSetMap			
d	LinkedMapSet			

Implemen	ntacija sučelja java.util.Set iz Javinog okvira kolekcija koja osigurava da su elementi sortirani je:
a	HashSet
b	ComparableSet
С	LinkedHashSet
d	TreeSet

Implementacija sučelja java.util.Set iz Javinog okvira kolekcija koja osigurava da su elementi poredani redoslijedom dodavanja je: TreeSet HashSet OrderedSet LinkedHashSet

lekcija nudi sljedeće implementacije sučelja ja	va.util.List:
List	
List	
yList	
cedList	
hList	
9	pList eList kedList



- b bilo koja klasa
- c primitivni tip
- d bilo koje sučelje
- samo klase koji su objektni omotači primitivnih tipova

svaki ključ ima pridruženu jednu vrijednost a mapa može pohranjivati više istih ključeva b mapa **ne može** pohranjivati više istih ključeva C više ključeva **ne smije** imati istu vrijednost d više ključeva može imati istu vrijednost е

Koje su karakteristike kolekcije Map?

Što od navedenog je ispravno ako je map tipa Map<String, Integer>?

```
for(String entry : map) {
   // radi nešto
}
```

```
for(String entry : map.keySet()) {
   // radi nešto
}
```

```
Sto od navedenog je ispravno ako je map tipa Map<String, Integer>?
```

```
a for(var entry : map) {
    // radi nošto
}
```

```
for(Map.Entry<String, Integer> entry : map.keySet()) {
    // radi neSto
}
```

```
for(Map.Entry<String, Integer> entry : map.values()) {
    // radi nesto
}
```

```
for(Map.Entry<String, Integer> entry : map) {
    // radi nešto
}
```

```
for(Map.Entry<String, Integer> entry : map.entrySet()) {
    // radi neSto
}
```

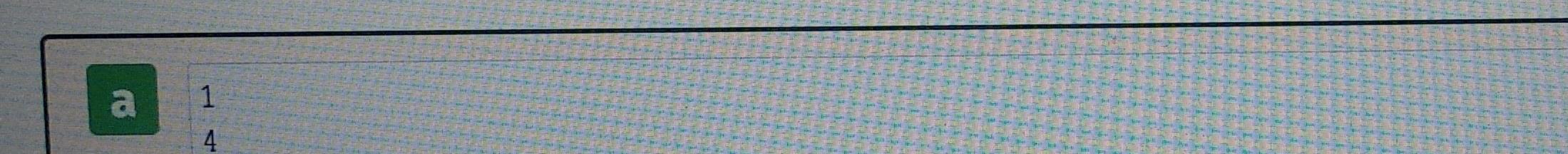
```
Sto sve od navedenog je ispravno ako je map tipa Map<String, Map<String, Integer>>?
          for(Map.Entry<String, Map.Entry<String, Integer>> entry : map.entrySet()) {
            // radi nešto
           for(Map.Entry<String, Map<String, Integer>> entry : map.entrySet()) {
             // radi nešto
           for(Map<String, Integer> entry : map.keySet()) {
             // radi nešto
           for(Map.Entry<Map<String, Integer>> entry : map.values()) {
     d
             // radi nešto
```

```
for(Map<String, Integer> entry : map.values()) {
e
```

// radi nešto

Ako je mapa definirana s Map<String, Integer> map ispravno inicijalizirana te je prazna u trenutku izvođenja sljedećih naredbi,

```
map.put("Ana", 5);
map.put("Ana", 4);
System.out.println(map.size());
System.out.println(map.get("Ana"));
```



što će se ispisati na ekranu?

```
Ako je skup definiran s Set<String> set ispravno inicijaliziran te je prazan u trenutku izvođenja sljedećih naredbi,
System.out.println(set.add("Ana"));
System.out.println(set.add("Ana"));
što će se ispisati na ekranu?
          false
          Ana
          false
          true
          true
          false
```

Metoda vratiKolekciju() vraća ne-null referencu na neku nepromjenjivu kolekciju. Što je ponuđenoga potrebno upisati svojem pozivatelju propagirala iznimku nastalu uslijed poziva metode add u bloku try?	na mjesto označeno podvlakama u kodu kako metoda m ne bi
<pre>public void m() {     try {         Collection<string> kolekcija = vratiKolekciju();         kolekcija.add("Pero");         System.out.println(kolekcija.contains("Pero"));     }     catch() { } }</string></pre>	
a IndexOutOfBoundsException ex	
b InvalidArgumentException ex	

NullPointerException ex

UnsupportedOperationException ex

```
Metoda vratiKolekciju() vraća referencu na neku kolekciju. Što treba upisati na mjesta označena u kodu kako bi program ispisao sve elemente kolekcije na zaslon?
 Collection<String> kolekcija = vratiKolekciju();
 for(___(1)___: ___(2)___) {
      System.out.println(___(3)___);
             (1) String str, (2) kolekcija, (3) iterator().next()
              (1) String str, (2) kolekcija.iterator(), (3) str
               (1) String str, (2) kolekcija, (3) iterator()
                 (1) String str, (2) kolekcija, (3) str
```

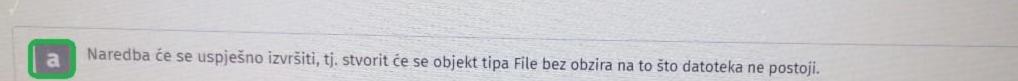
Napisali smo razre	ed koji implementira sucelje Filenan	neritter. U tom razreou sino i		
public boole	ean accept(File dir, String name) {	КОО }		to asset violeti \ fift
Što treba upisati	na mjesto linije " KOD" ako že	elimo provjeriti da li je objek	t iz datotečnog sustava za koji se po	ziya metoda attep of silo
a retu	rn new File(dir,name).isDirectory()	;		
b ret	urn name.endsWith(".");			
c re	eturn !(name.isFile());			
	return dir.isDirectory();			

od navedenog vrijedi pri preslikavanju primjeraka razreda File i Path:
primjerci razreda File mogu se preslikati u primjerke razreda Path koristeći metodu toPath(), a primjerci razreda Path se ne mogu preslikati u primjerke razreda File
U Javi ne postoje File i Path, već samo Files i Paths
primjerci razreda Path mogu se preslikati u primjerke razreda File koristeći metodu toFile(), a primjerci razreda File se ne mogu preslikati u primjerke razreda Path
primjerci razreda File ne mogu se preslikati u primjerke razreda Path, a ni primjerci razreda Path se ne mogu preslikati u primjerke razreda File
primjerci razreda File mogu se preslikati u primjerke razreda Path koristeći metodu toPath(), a primjerci razreda Path mogu se preslikati u primjerke razreda File koristeći metodu toFile()
֡֡֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜֜

Ako se	za obradu strukture direktorija oslonimo na metodu Files.walkFileTree(), jedan od objekata koji je potrebno predati može biti objekt koji je (označite sve što je točno):
a	primjerak klase koja implementira FileVisitor
b	primjerak klase koja implementira SimpleFileVisito
С	primjerak klase koja nasljeđuje FileVisitor
d	primjerak klase koja nasljeđuje SimpleFileVisitor

Ako se za obradu strukture direktorija oslonimo na metodu Files.walkF	ileTree(), označite sve što potrebno predati toj metodi kroz argumente:
objekt koji implementira sučelje FileVisitor	
b iterator koji vraća datoteke u direktoriju	
vršni direktorij iz kojeg kreće obilazak	
d objekt koji implementira sučelje DirectoryStream.Filter	

Pretpostavimo da koristimo računalo s Windows OS-om i da NE POSTOJI datoteka na putanji "d:/tmp/readme.txt". Što će se dogoditi ako pokrenemo program File f = new File("d:/tmp/readme.txt");



Pretpostavimo da koristimo računalo s Windows OS-om i da NE POSTOJI datoteka na putanji "d:/tmp/readme.txt". Što će se dogoditi ako pokrenemo program koji sadrži sljedeću liniju koda:

File f = new File("d:/tmp/readme.txt");



- Pogreška prevoditelja, ne postoji File konstruktor koji prima samo jedan parametar.
- Pogreška prevoditelja, datoteka ne postoji i kod se neće prevesti.
- Naredba će se uspješno izvršiti, a na navedenoj putanji će se stvoriti prazna datoteka readme.txt.
- Naredba će se uspješno izvršiti, tj. stvorit će se objekt tipa File, ali ne i datoteka readme.txt.
  - Kod će se prevesti ali dogodit će se pogreška pri izvođenju navedene linije, tj. "File not found exception".

Št	o sve vr	ijedi za kolekciju tipa <i>List</i>
1	a	može se dohvatiti element na zadanoj poziciji
	b	kolekcija je nepromjenjiva
	C	elementi se mogu dodavati samo na kraj
	d	može se umetnuti element na zadanu poziciju
	е	može se obrisati element na zadanoj poziciji

Što sve	vrijedi za kolekciju tipa <i>List</i>	
a	numeracija pozicija počinje od 1	
ь	može se dohvatiti element na zadanoj poziciji	
С	numeracija pozicija počinje od 0	
d	elementi nemaju svoju poziciju unutar liste	

elementi imaju svoju poziciju unutar liste

Ako se kod kreiranja novog FileInputStream objekta proslijedi kao parametar ime datoteke koja ne postoji, dogoditi će se: Bacit ĉe iznimka FileNotFoundException

Nad ob	jektima koji su primjerci razreda koji imp	plementira sučelje DirectoryStream, koje sve m	etode sigurno možemo pozvati (jer postoje)?
a	hasNext()		
E	iterator()		
	close()		