

Pismeni ispit iz Objektno orijentisanog programiranja i JAVA-e

1. U jednoj TV emisiji učestvuju voditelj Pera i gosti Svetlana, Sara, Simonida, Željko, Živko i Živan. Reditelj emisije je Zoran, pomoćnik reditelja je Zorana, dok je kamerman Radovan, a ton majstor Ratko. U publici se nalaze Milan, Milunka, Miroslav, Miroslava, Marko, Marijana, Milun i Milunka. Korsiteći metode klasifikacije, generalizaije i specijalizacije simulirati TV emisiju.

Emisija počinje tako što reditelj emisije Zoran kaže "kamera ton, idemo". U tom trenutku kamerman Radovan uključuje kameru Kamera i počinje snimanje, dok ton majstor Ratko u tom trenutku uključuje mikrofonski Mikrofon i počinje snimanje zvuka. Tog trenutka voditelj Pera izgovara tekst "Dobro došli u TV emisiju Jedna TV Emisija, pozdravimo današnje goste u studiju, Svetlanu Saru, Simonidu, Željka, Živka i Živana". Zatim daje neki kratak uvod u temu emisije (izaberite sami temu emisije) i pita nekog gosta (izaberite sami) šta misli o toj temi. Kad bude pitan gost odgovori na pitanje. Posle toga izaberete slučajni broj  $n \leq 6$ , i slučajno izaberete  $n$  gostiju koji nešto izgovaraju. Posle izjašnjavanja  $n$ -tog gosta nešto izgovori voditelj. Izaberite slučajni broj  $m$  i ponovite cikluse izjašnjavanja gostiju i voditelja  $m$  puta. Posle svakog izjašnjavanja gostiju ili voditelja reditelj na slučajni način odlučuje da li publika na govor reaguje smehom ili aplauzom, tako što obavesti pomoćnika da podigne tablu smeh ili aplauz. Po podizanju odgovarajuće table od strane pomoćnika reditelja svaki od ljudi u publici reaguje na odgovarajući način. Reditelj pre nego izjava voditelja ili gostiju biva emitovana može da izda naredjenje ton majstoru da je celokupnu zameni signalom "beep". Obezbediti da pre nego izjava voditelja ili gostiju bude odštampana na standardnom izlazu biva prvo na slučajni način izabrana od strane reditelja za zamenu signalom "beep". Po završetku  $m$  ciklusa izjašnjavanja voditelja i gostiju, reditelj daje objavu kraj. U tom trenutku snimatelj prestaje snimanje kamerom Kamera i ton majstor prestaje snimanje mikrofonom Mikrofon. Obezbediti da se tekst onoga što gosti i voditelj izgovaraju nalazi u klasi koja nasledjuje klasu `java.util.HashMap` i koja ima

metodu koja prihvata celobrojnu vrednost i vraća string koji predstavlja ono što vođitelj ili gost izgovara.

Potrebno je hijerarhiju klasu predstaviti UML-om.

Klasa koja se može koristiti kao generator slučajnih brojeva u JAVA-i je `java.util.Random`.

**Rešenje.** Jedan implementacija može biti sledeća

File `CovekUStudiju.java`

```
1 package Emisija;
3 import java.util.Random;
5 abstract class CovekUStudiju {
    private final String name;
    protected static Random rand = new Random();
9    protected CovekUStudiju(String name) {
        this.name = name;
11    }
13    public String getName() {
        return name;
15    }
}
```

File `Govornik.java`

```
package Emisija;
2
3 abstract class Govornik extends CovekUStudiju {
4     private final Tekstovi tekstovi;
        private static final char[] symbols;
6     protected int brojGovora;
8     static {
        StringBuilder tmp = new StringBuilder();
10        for (char ch = '0'; ch <= '9'; ++ch)
            tmp.append(ch);
12        for (char ch = 'a'; ch <= 'z'; ++ch)
            tmp.append(ch);
14        symbols = tmp.toString().toCharArray();
16    }
}
```

```

18         protected Govornik(String name, int brojGovora) {
19             super(name);
20             this.brojGovora = brojGovora;
21             tekstovi = new Tekstovi(brojGovora);
22             for(int i = 0; i < brojGovora; ++i)
23                 tekstovi.put(i, getRandomString(rand.nextInt
24                 (30)));
25         }
26
27         public String govori() {
28             return tekstovi.get(rand.nextInt(brojGovora));
29         }
30
31         private String getRandomString(int length) {
32             char buf[] = new char[length];
33             for(int i = 0; i < length; ++i)
34                 buf[i] = symbols[rand.nextInt(symbols.length)];
35             return new String(buf);
36         }
37     }

```

#### File Gost.java

```

1 package Emisija;
2
3 public class Gost extends Govornik {
4
5     public Gost(String name, int brojGovora) {
6         super(name, brojGovora);
7     }
8 }

```

#### File Voditelj.java

```

1 package Emisija;
2
3 public class Voditelj extends Govornik {
4     public String otvoriEmisiju() {
5         return "Dobro dosli u emisiju Jedna TV Emisija, pozdravimo
6             danasnje goste u studiju, Svetlanu, Saru, Simonidu, Zeljka,
7             Zivka i Zivana.";
8     }
9 }

```

```

8         public String temaEmisije() {
          return "Tema danasnje emisije je ko zna sta, mada ni
10         to nije tako neobicno, sta mislite vi Svetlana";
        }

12         public String odjava() {
          return "Eto dragi gledaoci lepo smo popricali sa
14         nasim gostima na temu ko zna sta. Dovidjenja i do gledanja.";
        }

16     public Voditelj(String name, int brojGovora) {
18         super(name, brojGovora);
    }
}

```

#### File GostUPublici.java

```

1 package Emisija;

3 public class GostUPublici extends CovekUStudiju {

5     public GostUPublici(String name) {
6         super(name);
7     }

9     public void reaguj(String reakcija) {
10        System.out.println(getName() + " : " + reakcija);
11    }
}

```

#### File PomocnikReditelja.java

```

1 package Emisija;

2 public class PomocnikReditelja extends CovekUStudiju {
3     private Tabla tabla;

4     public PomocnikReditelja(String name, Tabla tabla) {
5         super(name);
6         this.tabla = tabla;
7     }

8     public void aplauz() {
9
10    }
}

```

```

12     tabla.prikazi("aplauz");
13         tabla.ukljuci();
14 }
15
16 public void smeh() {
17     tabla.prikazi("smeh");
18     tabla.ukljuci();
19 }
20
21     public void iskljuciTablu() {
22         tabla.iskljuci();
23     }
24 }

```

#### File Reditelj.java

```

package Emisija;

2
public class Reditelj extends CovekUStudiju {
4     private TonMajstor tonMajstor;
5     private PomocnikReditelja pomocnikReditelja;
6     private Kamerman kamerman;

8     public Reditelj(String name, TonMajstor tonMajstor, Kamerman
        kamerman, PomocnikReditelja pomocnikReditelja) {
9         super(name);
10         this.tonMajstor = tonMajstor;
11         this.kamerman = kamerman;
12         this.pomocnikReditelja = pomocnikReditelja;

14     }

16     public void pristojno(String izgovoreno) {
17         if(!rand.nextBoolean())
18             tonMajstor.emitujBeep();
19         else
20             tonMajstor.emituj(izgovoreno);
21     }

22     public void reakcija() {
23         int i = rand.nextInt(10);
24         if(i<2)
25             pomocnikReditelja.aplauz();
26         if(i<4)

```

```

28         pomocnikReditelja.sneh();
           else
30         pomocnikReditelja.iskljuciTablu();
    }
32
    public void kreni() {
34        System.out.println("Kamera ton, kreni");
    }
36
    public void stani() {
38        tonMajstor.stopMikrofon();
           kameraman.stopKamera();
40    }
}

```

#### File Kamerman.java

```

1 package Emisija;

3 public class Kamerman extends CovekUStudiju {
    private Kamera kamera;
5
    public Kamerman(String name, Kamera kamera) {
7        super(name);
           this.kamera = kamera;
9    }

11    public void startKamera() {
        kamera.ukljuci();
13    }

15    public void stopKamera() {
        kamera.iskljuci();
17    }
}

```

#### File TonMajstor.java

```

package Emisija;

2
public class TonMajstor extends CovekUStudiju {
4    private Mikrofon mikrofon;

```

```

6   public TonMajstor(String name, Mikrofon mikrofon) {
      super(name);
8       this.mikrofon = mikrofon;
    }
10
11   public void startMikrofon() {
12       mikrofon.ukljuci();
13   }
14
15   public void stopMikrofon() {
16       mikrofon.iskljuci();
17   }
18
19       public void emitujBeep() {
20           System.out.println("beep");
21       }
22
23       public void emituj(String str) {
24           System.out.println(str);
25       }
26 }

```

#### File Uredjaj.java

```

1   package Emisija;
2
3   abstract class Uredjaj {
4       private boolean ukljucen;
5
6       protected Uredjaj() {
7           ukljucen = false;
8       }
9
10      public void ukljuci() {
11          ukljucen = true;
12      }
13
14      public void iskljuci() {
15          ukljucen = false;
16      }
17
18          public boolean ukljucen() {
19              return ukljucen;
20          }

```

```
}
```

#### File Kamera.java

```
1 package Emisija;  
3 public class Kamera extends Uredjaj {  
5 }
```

#### File Mikrofon.java

```
1 package Emisija;  
3 public class Kamera extends Uredjaj {  
5 }
```

#### File Tabla.java

```
1 package Emisija;  
3 public class Tabla extends Uredjaj {  
    private String sadrzaj;  
5  
    public void prikazi(String str) {  
6        sadrzaj = str;  
7        //System.out.println("Tabla: " + str);  
9    }  
11  
    public String sadrzaj() {  
12        return sadrzaj;  
13    }  
15 }
```

#### File Tekstovi.java

```
1 ackage Emisija;  
3 import java.util.HashMap;
```



```

5 class Tekstovi extends HashMap<Integer , String> {
7     public Tekstovi(int i) {
9         super(i);
11    }

```

#### File JednaTVEmisija.java

```

1 package Emisije;
3 import Emisija.Voditelj;
4 import Emisija.Gost;
5 import Emisija.GostUPublici;
6 import Emisija.Reditelj;
7 import Emisija.TonMajstor;
8 import Emisija.Kamerman;
9 import Emisija.PomocnikReditelja;
10 import Emisija.Kamera;
11 import Emisija.Mikrofon;
12 import Emisija.Tabla;
13 import java.util.Random;
14 import java.util.Vector;
15
16 public class JednaTVEmisija {
17     private Voditelj pera;
18     private Vector<Gost> gosti;
19     private Vector<GostUPublici> publika;
20     private Reditelj zoran;
21     private TonMajstor ratko;
22     private Kamerman radovan;
23     private PomocnikReditelja zorana;
24     private Kamera kamera;
25     private Mikrofon mikrofon;
26     private Tabla tabla;
27     private int brojGovoraVoditelja;
28     private Random rand;
29
30     public JednaTVEmisija() {
31         pera = new Voditelj("Pera",30);
32         gosti = new Vector<>(6);
33         gosti.add(new Gost("Svetlana",20));

```

```

35      gosti.add(new Gost("Sara",20));
36      gosti.add(new Gost("Simonida",10));
37      gosti.add(new Gost("Zeljko",10));
38      gosti.add(new Gost("Zivko",13));
39      gosti.add(new Gost("Zivan",21));
40      publika = new Vector<>(6);
41      publika.add(new GostUPublici("Milan"));
42      publika.add(new GostUPublici("Milanka"));
43      publika.add(new GostUPublici("Miroslav"));
44      publika.add(new GostUPublici("Miroslava"));
45      publika.add(new GostUPublici("Marko"));
46      publika.add(new GostUPublici("Marijana"));
47      publika.add(new GostUPublici("Milun"));
48      publika.add(new GostUPublici("Milunka"));
49
50      kamera = new Kamera();
51      mikrofon = new Mikrofon();
52      tabla = new Tabla();
53      zorana = new PomocnikReditelja("Zorana",tabla);
54      radovan = new Kamerman("Radovan", kamera);
55      ratko = new TonMajstor("Ratko",mikrofon);
56      zoran = new Reditelj("Zoran",ratko, radovan,
57      zorana);
58
59      rand = new Random();
60      brojGovoraVoditelja = 3 + rand.nextInt(10);
61  }
62
63  public void emituj() {
64      zoran.kreni();
65      System.out.println("Pera :");
66      System.out.println(pera.otvoriEmisiju());
67      System.out.println(pera.temaEmisije());
68      zoran.reakcija();
69      System.out.println(gosti.get(0).getName() + " :
70      ");
71
72      zoran.pristojno(gosti.get(0).govori());
73      zoran.reakcija();
74      for(int i = 0; i < brojGovoraVoditelja; ++i) {
75          if(i>0) {
76              System.out.println("Pera :");
77              zoran.pristojno(pera.govori());
78          }
79          int govorGostiju = 2 + rand.nextInt(6);
80          for(int j = 0; i < govorGostiju; ++i) {
81              int gost = rand.nextInt(6);

```

```

77         System.out.println(gosti.get(gost).
getName() + " : ");
79         zoran.pristojno(gosti.get(gost).govori()
);
81         zoran.reakcija();
            if(tabla.ukljucen()) {
                for(int k = 0; k < 8; ++k)
                    publika.get(k).reaguj(tabla.
sadrzaj());
83             }
            }
85         }
        System.out.println(pera.odjava());
87         zoran.stani();
    }
89     public static void main(String argv[]) {
        JednaTVEmisija jednaTVEmisija = new JednaTVEmisija()
;
91         jednaTVEmisija.emituj();
93     }
}

```

UML dijagram ja dat u posebnom fleu.

2. Koristeći UML diagram Napisati klasu A, pretopostavljajući da je deo paketa **rs.ac.bg.mas**.

**Rešenje.** Rešenje zadatka može biti sledeća klasa

File **rs/ac/bg/mas/A.java**

```

1  package rs.ac.bg.mas;
3  public class A {
5      String a;
7      public A() {
            a="a";
9      }
11     public A(String a) {
            this.a=a;
13     }
15     public interface I {
            public void method1(String a);

```

```
17         public void method2(String a);
18     }
19
20     public class B implements I {
21         private String b;
22         @Override
23         public void method1(String a) {
24             b="1";
25         }
26
27         @Override
28         public void method2(String a) {
29             b="2";
30         }
31     }
32
33     public I ai = new I() {
34         private String b;
35         @Override
36         public void method1(String a) {
37             b=a;
38         }
39
40         @Override
41         public void method2(String a) {
42             b=a;
43         }
44     };
45
46     public static void main(String argv[]) {
47
48     }
49 }
```

## Class Diagram2

