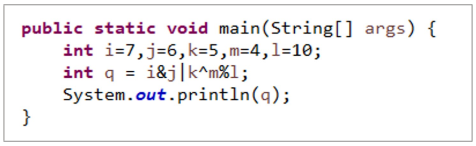
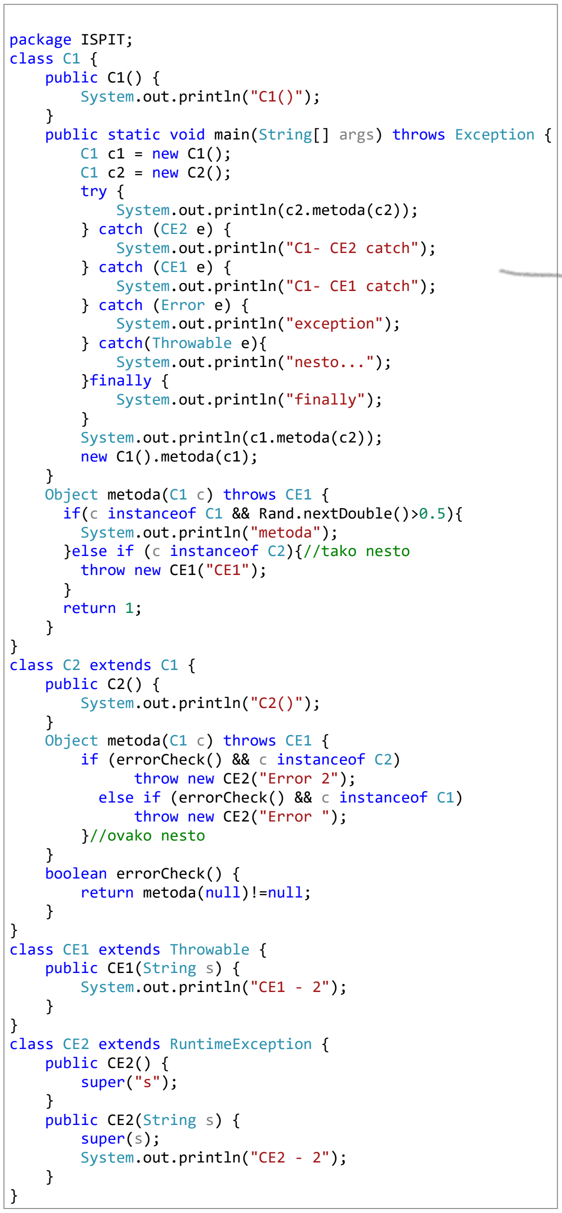
Monday, May 19, 2025 2:55 PM

T40

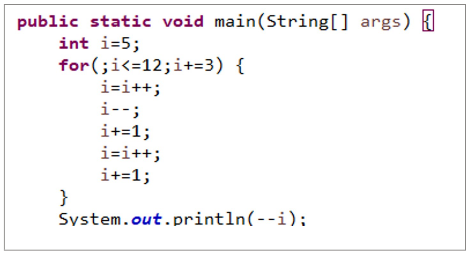


```
public static void main(String[] args) {
    int i=7,j=6,k=5,m=4,l=10;
    int q = i&j|k^m%l;
    System.out.println(q);
}
```



```
package ISPIT;
class C1 {
    public C1() {
        System.out.println("C1()");
    }
    public static void main(String[] args) throws Exception {
        C1 c1 = new C1();
        C1 c2 = new C2();
        try {
            System.out.println(c2.metoda(c2));
        } catch (CE2 e) {
            System.out.println("C1- CE2 catch");
        } catch (CE1 e) {
            System.out.println("C1- CE1 catch");
        } catch (Error e) {
            System.out.println("exception");
        } catch(Throwable e){
            System.out.println("nesto...");
        }finally {
            System.out.println("finally");
        }
        System.out.println(c1.metoda(c2));
        new C1().metoda(c1);
    }
    Object metoda(C1 c) throws CE1 {
      if(c instanceof C1 && Rand.nextDouble()>0.5){
        System.out.println("metoda");
      }else if (c instanceof C2){//tako nesto
        throw new CE1("CE1");
      }
      return 1;
    }
}
class C2 extends C1 {
    public C2() {
        System.out.println("C2()");
    }
    Object metoda(C1 c) throws CE1 {
        if (errorCheck() && c instanceof C2)
              throw new CE2("Error 2");
          else if (errorCheck() && c instanceof C1)
              throw new CE2("Error ");
        }//ovako nesto
    }
    boolean errorCheck() {
        return metoda(null)!=null;
    }
}
class CE1 extends Throwable {
    public CE1(String s) {
        System.out.println("CE1 - 2");
    }
}
class CE2 extends RuntimeException {
    public CE2() {
        super("s");
    }
    public CE2(String s) {
        super(s);
        System.out.println("CE2 - 2");
    }
}
```

→ bile 2 kompajlerske greške!



```
public static void main(String[] args) {
    int i=5;
    for(;i<=12;i+=3) {
        i=i++;
        i--;
        i+=1;
        i=i++;
        i+=1;
    }
    System.out.println(--i);
```

```java
package usmeni.Z5;

import java.io.Serializable;
public class A1 {
    static {
        System.out.println("A1-S");
    }
    {
        System.out.println("A1-N");
    }
    private A1 a1;
    private A1() {
        System.out.println("A1()");
    }
    public A1(A1 a1) {
        this();
        System.out.println("A1(A1)");
        this.a1 = a1;
        new A2(a1);
    }
    public void metoda1() {
        System.out.println("A1.metoda1()");
    }
}
class A2 extends A1 implements Serializable {
    protected A2() {
        System.out.println("A2()");
    }
    public A2(A1 a1) {
        System.out.println("A2(A1)");
        a1.metoda1();
    }
    public void metoda1() {
        System.out.println("A2.metoda1()");
    }
    public void metoda2() {
        System.out.println("A2.metoda2()");
    }
}
class A3 extends A2 {
    A2 a2 = null;
    static {
        System.out.println("a3-S");
    }
    {
        System.out.println("a3-N");
    }
    public A3() {
        super();
        System.out.println("a3()");
    }
    public A3(A2 a2) {
        this.a2 = a2;
        System.out.println("a3(A2)");
    }
    public A3(A1 a1, A2 a2) {
        this(a2);
        System.out.println("a3(A1,A2)");
    }
    public void metoda2() {
        System.out.println("a3.metoda()");
    }
}
class A4 extends A3 {
    A1 a1 = new A1();
    A3 a2 = new A3(new A1(new A1()), new A2(a1));
    Serializable a3 = new A3();
    public A4() {
        super();
        System.out.println("A4()");
        super.metoda1();
    }

    public static void main(String[] args) {
        A4 a4 = new A4();
        a4.metoda1();
        a4.metoda2();
        ((A2) a4).metoda1();
        ((A2) a4.a3).metoda2();
        Serializable a5 = new A5();
        ((A3) a5).metoda2();
    }
}
class A5 extends A2 {
    A4 a4 = new A4();
    static {
        System.out.println("A5-S");
    }
    public void metoda2() {
        System.out.println("A5.metoda2()");
    }
    public A5() {
        System.out.println("A5()");
        metoda1();
    }
    static {
        ((A1) new A2(new A1())).metoda1();
    }
}
```

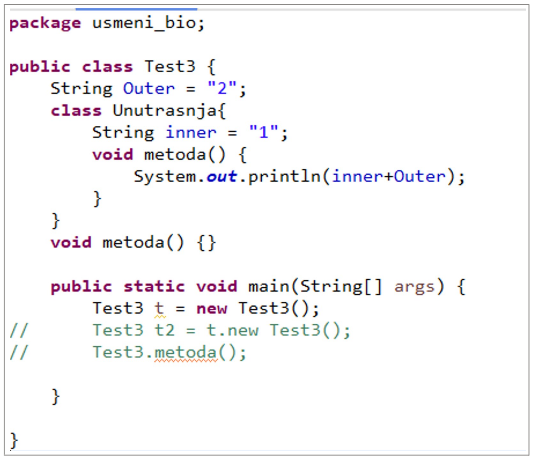
→ nešto ovako, static imao samo System.out.println(...)

private A1()

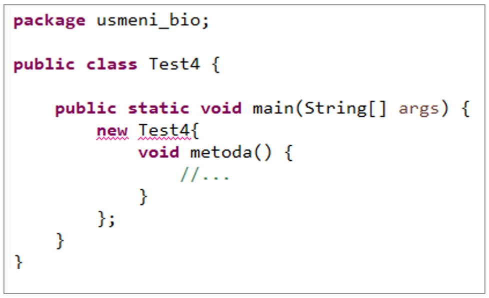
bio

Greška je bila druga.

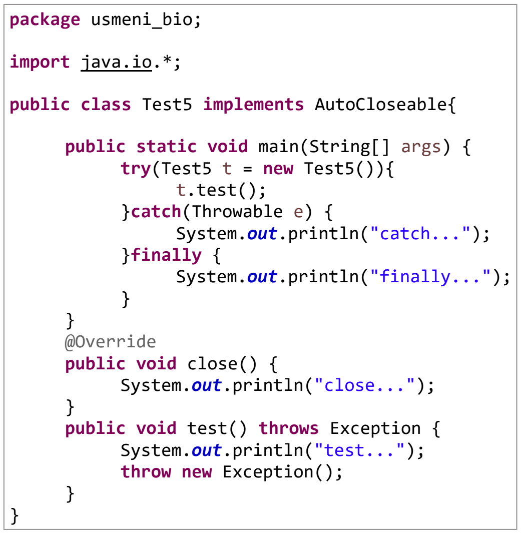
Ako se dobije neki sa private konstruktorom, pogledati da li postoje druge kompajlerske greške!



```
package usmeni_bio;

public class Test3 {
    String Outer = "2";
    class Unutrasnja{
        String inner = "1";
        void metoda() {
            System.out.println(inner+Outer);
        }
    }
    void metoda() {}

    public static void main(String[] args) {
        Test3 t = new Test3();
//      Test3 t2 = t.new Test3();
//      Test3.metoda();

    }

}
```



```
package usmeni_bio;

public class Test4 {

    public static void main(String[] args) {
        new Test4{
            void metoda() {
                //...
            }
        };
    }
}
```



```
package usmeni_bio;

import java.io.*;

public class Test5 implements AutoCloseable{

      public static void main(String[] args) {
            try(Test5 t = new Test5()){
                  t.test();
            }catch(Throwable e) {
                  System.out.println("catch...");
            }finally {
                  System.out.println("finally...");
            }
      }
      @Override
      public void close() {
            System.out.println("close...");
      }
      public void test() throws Exception {
            System.out.println("test...");
            throw new Exception();
      }
}
```

```
package test2;

public class G1 {
      public static void main(String[] args) {
            G3 g3 = new G3();
            GT1<G3, G2> gt1 = new GT1<G3, G2>();
            GTI<Integer> gt2 = new GT1<>()//na ovaj tip
zadatak,isti,iste klase, samo je izmenjen G3 klasa necim…,I parametri,
nije Integer vec float, ili nije GTI<> vec <Object,Integer>…
            G2 g2 = new G2("g2");
            GTI<Integer> gt3 = new GT1<>();
            gt3.method(2);
            g3.method(2f);
            gt3.method();
            g3.add(gt3.method());
            g3.method2(3);
            gt1.method(g2);
            System.out.println(gt1.t.method());
            gt1.t.method("gt1");
            System.out.println(gt1.t.method());
            g2.method("g22");
            System.out.println(g3.method());
            System.out.println(gt1.t.method());
            System.out.println(gt2.method());
            System.out.println(gt3.method());
      }
}

class GT1<T, T3> implements GTI<T3> {
      T3 t;

      public void method(T3 t) {
            this.t = t;
      }

      public T3 method() {
            System.out.println(t);
            return t;//mozda, nije bilo vrati null
      }
}

class G2 extends Exception {
      String x = "G2";

      G2(String x) {
            x = x;
      }

      void method(String a) {
            x = a;
      }

      String method() {
            return x;
      }
}

class G3 extends GT1<Integer, Float> implements GTI2<Integer> {
      public void add(Integer i) {
            System.out.println("Class 2: " + t.getClass());
            t += i;
      }

      public void method2(Integer t) {
            this.t += t++;
      }
}

interface GTI<T2> {
      public void method(T2 t);

      public T2 method();
}

interface GTI2<T1> {
      public void method2(T1 t);
}
```

```java
package usmeni_bio;

public class Test6 implements I1{
	public static void main(String[] args) {
		//nesto poput
		//I1 i = (c)->System.out.println(c*2);
	}
}

interface I1{
	int hashCode();
	public abstract boolean equals();
	public void add(double x);
	public void metoda(int nesto);
}

interface I2{//mozda extends I1
	public void doSmth();
}
```

```java
package usmeni_bio;

public class Test7 {
	Integer A;
	int a;

	public Test7(int a) {
		this.a=A+a;
	}
	public static void main(String[] args) {
		Integer A = new Integer(5);
		new Test7(A);
	}
}
```

+ Nešto na ovaj fazon:

```java
package usmeni.primjeri2.T2;
/*public class T2 {
    private void m1() {
        System.out.println("1");
    }
    public static void main(String[] args) {
        new TA() {
             void m1() {
                System.out.println("1");
            }
        }.m2();
    }
}
interface TI {
    void m1();
    void m2();
}
abstract class TA implements TI {
    public void m2() {
        System.out.println("2");
    }
}*/
```

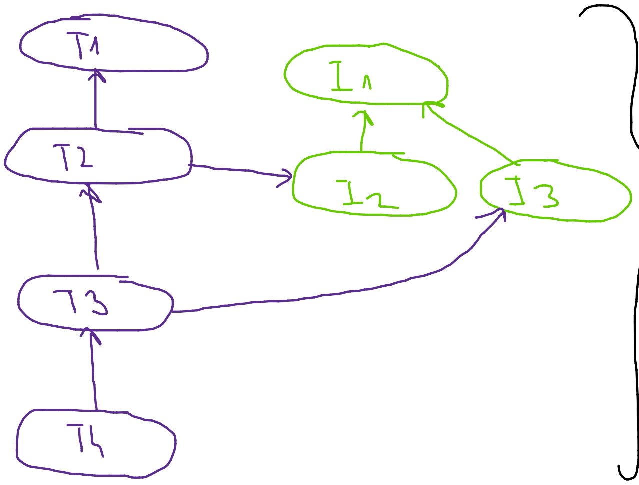
```java
package usmeni_bio;

public class T9 {

	public static void main(String[] args) {
		int i=4;
		i = i++ % 6;
		System.out.println(i);
		i = i++ % 5;
		System.out.println(i);
		i = i++ % 4;
		System.out.println(i);
	}

}
```

Obratiti pažnju:

🟪 → boja-klasa

🟩 → interfejs

4. zadatak bio



PRI NASLEĐIVANJU, → (metoda), .metoda (metoda3()) bila public u I3, a package-private u T3...

DOŠLO DO SMANJENJA VIDLJIVOSTI, iz public u package-private

ima dovoljno vremena, svaki zadatak pregledati za ovo (isplati se)