

Név	Neptun	1	2	3	4	Σ	Jegy

Rendelkezésre álló idő: **120 perc**

1. Mit ír ki az alábbi program?

15 pont

- Az alábbi forráskód egy `Mitirki.java` nevű file-ban van elhelyezve.

```
public class Mitirki {
    public static void main(String[] args) {
        Leszarmazott o = new Leszarmazott();
        Os o2 = o;

        System.out.println(o.x);
        System.out.println(o2.x);
    }
}

class Os {
    static {
        System.out.println("a");
    }

    {
        System.out.println("b");
    }

    public int x = 3;

    public Os() { System.out.println("os1"); }
    public Os(int x) { this(); System.out.println("os2"); }
}

class Leszarmazott extends Os {
    public String x = "abcd";

    public Leszarmazott() { super(3); System.out.println("leszarmazott"); }
}
```

2. Keresd meg a fordítási hibákat az alábbi kódban!

15 pont

- Az alábbi forráskód egy `ForditasiHiba.java` nevű file-ban van elhelyezve.
- A fordítás a `javac -classpath . ForditasiHiba.java` paranccsal történik, amelynek kiadásakor a forráskódot tartalmazó könyvtár az aktuális.
- Add meg a hibák helyét (sorszám), és okát!

```
1 public class ForditasiHiba extends Os, I {
2     static {
3         int a = 0, b= 1;
4         init( a+b );
5     }
6     public void init( int i) {
7         System.out.println( i );
8     }
9     public long osbenDefinialtMetodus()
11        System.out.println("hihi");
12    }
13    public void sajátMetodus() {
14        this.OsMetodus();
15    }
16}
17
18interface I {
19    private int privatAdat = 0;
20    void f() { System.out.println("Hello");
21}
22
23class Os {
24    int osbenDefinialtMetodus() {
25        System.out.println("hoho");
26    }
27    private int OsMetodus() throws Exception {
28        System.out.println( "alma" );
29        return 3;
30    }
31}
32
33public static void main( String[] args ) {
34    ForditasiHiba.osbenDefinialtMetodus();
35}
```

3. Ismertesse a JVM osztálybetöltési algoritmust (delegáció)! Milyen struktúrába szerveződnek az osztálybetöltők? Ha hivatkozunk egy osztályra, melyik osztálybetöltőhöz fordulunk? Hogyan fordulhat elő, hogy egy osztálynak több példánya szerepel a rendszerben és ez milyen problémákat okozhat?

30 pont

4. Készítse el a következő alkalmazást:

40 pont

- Szöveges fájlból (source) eseményeket olvasunk (minden sor egy esemény), amelyek két értékből állnak: időbélyeg (long timestamp) és érték (int).
- A feladat az események időhelyes visszajátszása (az első esemény mehet azonnal, de a többinél az események időbélyege közti különbséget ki kell várni).
- Property fájlból olvassa be a szöveges fájl elérési útját!
- Az esemény kiolvasását tekintheti atomi eseménynek (zérus késleltetést okoz).
- Amikor egy eseményt visszajátszunk, juttassuk el egy feldolgozó egységnek (sink). A feldolgozó egység (thread safe) nem zérus ideig blokkolni fog, ezért hívjuk külön szálból, hogy a visszajátszás időhelyességét ne rontsa el!

Javasolt architektúrában az adat áramlása (érdeemes ezt a 3 osztályt megvalósítani és a main függvényben összekötni őket):

Source (szöveges fájl) ----> Player -----> Sink (nem zérus feldolgozási idővel)