

# Többszálas prímszámkeresés csövekkel

Párhuzamosítsd az első órai prímszámkereső algoritmust, ami megkeresi egy adott  $n$  számnál kisebb prímszámokat! A program működési elve a következő:

- Az első szál sorban küldi a prímszámokat a második szálnak.
- A második szál kiírja a megkapott prímszámot, majd annak többszöröseit kiszűri, a megmaradt számokat pedig továbbküldi a harmadik folyamatnak.
- A 3... $k$ -ik folyamat hasonlóan működik, mint az második. Tehát mindenki a rákövetkező szálnak küld prímszámokat.

A szálak közötti kommunikációt a `java.io.PipedInputStream` és `java.io.PipedOutputStream` segítségével valósítsd meg!

Érd el, hogy az alkalmazás megfelelően leálljon, mikor az utolsó intervallumba eső prímszám is feldolgozásra került!

Érdemes lehet a szálak számát korlátozni, erre egy megoldás lehet, ha az utolsó szál már nem küldi tovább a feldolgozandó számokat, hanem ő maga dolgozza föl azokat.

Nem túl gazdaságos sok szál indítani. Mi lenne, ha az első szál az első 100 primmel filterezne, a második szál az utána következő 1000 primmel, a harmadik az utána következő 10000-rel, stb. Végezz méréseket, hogy vajon milyen konstansokat érdemes használni!