

Korszerű Programozási Technikák

KisZH IV - Pótlás

Adott a **Pair** osztály, amely egy egész számokból álló számpárt reprezentál, két int adattaggal (*a*, *b*). Az osztálynak van egy **negative** metódusa, ami **true**-val tér vissza ha az $a - b$ különbség negatív, **false**-el ha nem, továbbá van egy **print** függvény, ami kiírja a konzolra a számpárt. A **main.cpp**-ben a **generateData** metódus véletlenszerűen generál párokat és feltölt velük egy listát. A feladat szál kezeléshez és a lambda kifejezésekhez kapcsolódik: sok-sok számpárt kell ellenőrizni és az kiíratni párhuzamosan, több szálon. Elég csak a **main.cpp**-be dolgozni, figyelj a szálak megfelelő visszacsatolására.

1. Készíts egy **count** függvényt, ami paraméterben megkapja a számpárok listáját és egy **int*** mutatót. Számold meg, hogy hány olyan páros van, aminek negatív a különbsége és állítsd be a mutató értékét erre a darabszámmra. Minden iterációban aludjon a szál 1 milliszekundumot és írja ki, hogy addig hány számpárt ellenőrzött már. (1 pont)
2. Írj egy lambda függvényt a **main**-be, ami a **capture-list**-ben megkapja a **negative_count**-ot, a függvény paramétere a számpár lista, visszatérési értéke pedig **void**. A függvénytörzsben a **count** segítségével állítsd be a **negative_count** értékét a negatív különbségű párok darabszámára. A lambda függvényt indítsd el egy külön szálon a **main** függvényben a megfelelő paraméterekkel. (2 pont)
3. Írj egy **printPairs** nevű metódust, ami megkapja a **pairs** listát, **range-based loop** használatával végig megy a listán és kiírja a számpárokat. Minden kiíratás előtt aludjon a szál 1 milliszekundumot. A **main** metódusban indítsd el a **printPairs**-t egy új szálon a megfelelő paraméterekkel. (1 pont)
4. A **printPairs** és a **count** függvény is végez kiíratást a konzolra. Egészítsd ki a két függvényt (a lambda kifejezést, szálak indítását is módosítani kell) kölcsönös kizárás segítségével, hogy a kiíratások ne keveredjenek, mindegyik külön sorban jelenjen meg. (1 pont)