

Java programozás 2. ZH

Figyelem, 100 pont elérése számít 100%-os eredménynek!

A feladatokat figyelmesen olvasd végig, tervezd meg a kódot és csak ezután készítsd el. A programnak futnia kell és a kódnak a kivételeket kezelnie kell. A ZH írása során kizárólag a **csatolt hálózati meghajtón** található anyagok használhatók. Meg nem engedett segédeszköz használata az aláírás megtagadását vonja maga után.

Időmegtakarítás végett JavaDoc NEM szükséges a teljes pontszám eléréséhez.

Egyetlen projektet készíts! Az egyes feladatoknak külön csomagja legyen:

***hu.ppke.itk.java.DIGITUSAZONOSÍTÓ.zh2.feladatN*, ahol a DIGITUSAZONOSÍTÓ a digitus azonosítód, a feladatN-ből az N pedig a feladat sorszáma.**

1. feladat Aknamező szimuláció (65%)

A feladat egy aknamező szimulációjának a létrehozása. Két ellenséges csapat áll szemben a mezőn, az egyik aknákat helyez el, a másik pedig aknákat szed fel. A feladat a folyamatot szimulálni és a GUI-n megjeleníteni.

A szimuláció leírása

Aknamező

Adott egy $n \times m$ méretű négyzetrács, amely mezőit *celláknak* nevezzük. Egy cella állapota lehet

- üres
- aknásított
- aknaszedő dolgozik
- aknalerakó dolgozik

Értelemszerűen az aknaszedő az aknásított mezőről szed le aknát, míg az aknalerakó üres mezőt aknásít.

Az egyes cellákat egy `Cell` osztály példánya reprezentálja, amelyeket egy megfelelő Collection-ben kell elhelyezni. Ezekhez a cellákhoz fér hozzá a `MineLayer` (aknalerakó), valamint a `MineSweeper` (aknaszedő).

Aknaszedő és aknalerakó

Aknaszedőből és aknalerakóból egyaránt 5-5 dolgozhat az aknamezőn. Továbbá az is igaz, hogy lehetőség szerint mindig tartalékban van egy-egy szakember. Ez azt jelenti, hogy az 5-5 emberből 4-4 dolgozik folyamatosan, a tartalékosok várakoznak.

A tartalékos aknaszedőt akkor kell bevetni, ha az aknamező 50%-a el van aknásítva (tehát az aknalerakók vannak előnyben.) A tartalékos aknalerakót akkor kell mozgósítani, ha az aknamezőn már csak 10% az aknák száma (tehát az aknaszedők előnye látványos.) A tartalékosok azonnal pihenni térnek, ha a kritérium alá (50% alá), illetve fölé (10% fölé) kerül az aknák száma.

A terep változó, ezért egy aknával (mind lerakáskor, mind felszedéskor) 500 ms időt kell eltölteni. Továbbá két munka között szintén eltelik 200 ms idő.

Egy mezőn egyszerre egyvalaki dolgozik.

A szimuláció indítása után az aknamező üres, tehát eleinte az aknaszedőknek nincsen dolguk, a tartalékos aknalerakó viszont dolgozik (mert még 10% alatt van az aknák száma). (Ez következik az előzőkből, csak a tisztázás miatt írtuk le még egyszer.)

A programozási feladat részletei

Az aknaszedők és az aknalerakók szálak, amelyek az aknamezőn párhuzamosan dolgoznak. Az aknamező feladata „értesíteni” a tartalékosokat arról, hogy akcióba kell-e lépniük, vagy sem.

A szakemberek véletlenszerűen választanak egy cellát, és ha az foglalt, akkor várakoznak annak felszabadulásáig. Ha olyan cellát választott, ahol azonos foglalkozású szakember található, akkor természetesen arra nem várakozik. Ha olyan cellát választott, amelyen nincsen munka (tehát az aknalerakó ott aknát talál, valamint az aknaszedő üres mezőre sétált) akkor is időbe telik a következő mező meghatározása (tehát csak a munkaidőt lehet és kell elhagyni.)

GUI

A feladat része a megjelenítés is, ahol a képernyőre GUI-n kell mutatni az aknamező állapotát. Fontos, hogy a megjelenítés legyen szálbiztos! (Tehát legyen egy négyzetrács, amin a szimuláció nyomon követhető. Vezérlés nem kell.)

Mivel szálak vannak, mindenképpen ügyelj a szálbiztosságra és a hatékonyságra. (Ne legyen busy waiting vagy polling!)

2. feladat - POP3 parancssori kliens (35%)

A POP3 egy protokoll, amelyen keresztül a levelezőkliensek letöltik a levelezőszerverről az újonnan érkezett leveleket.

Készíts POP3-at használó, működő levelezőkliens, amivel egyszerű módon le tudod kérdezni a leveleidet a szervertől. A zhfeladat mappában megtalálható egy egyszerű szerver, amit használva ki tudjátok próbálni a működését. A projektet importálva a szerver elindítható.

Hibakezeléssel nem kell foglalkozni: ha hibaüzenet érkezik, akkor írja ki azt a konzolra!

A POP3 protokoll működése:

- A szerverhez a 110-es TCP porton kell kapcsolódni
- A szerverrel egyszerű szöveges üzenetekkel lehet kommunikálni, melyek CRLF-fel vannak lezárva (azaz egy 13-as és egy 10-es karakterkódú sortörés, Ezeket kell beírni Java-ban egy stringbe a CRLF-hez: „\r\n”)
- Minden üzenet egy kulcsszóból áll, melyet kis- és nagybetű függetlenül ki lehet adni, ez után pedig opcionálisan paraméterek állhatnak
- A szerver minden üzenetre választ ad, melyek az alábbi módon nézhetnek ki:
 - +OK Üzenet
 - -ERR Üzenet
- Az üzenetek tartalma sok esetben csak tájékoztató jellegű, különböző szervereken más található ott

A kommunikáció folyamata

- Kapcsolódni kell a szerverhez a 110-es porton
- Erre válaszul a szerver üdvözlő üzenetet küld:
 - +OK Successfully connected, you can log in now
- A bejelentkezéshez a következő adatokat kell használnod:
 - Felhasználónév: digitusos azonosítód
 - Jelszó: NEPTUN kódod, nagybetűvel
- Felhasználónév megadása
 - USER felhasználónév
- Szerver válaszol
 - +OK Great, send me your Neptun code
 - -ERR No such user
- Jelszó megadása
 - PASS jelszó
- Szerver válaszol
 - +OK Successful login, have a nice day!
 - -ERR Authentication failed, wrong Neptun code!

Amennyiben sikeres volt a belépés, a következő funkciók hajthatók végre:

- Üzenetek számának lekérdezése
 - Kérés: STAT
 - Válasz: +OK <üzenetszám> <üzenetek méret bájtban>
 - Üzenet lekérdezése:
 - Kérés: RETR <üzenetsorszám>
 - Ahol <üzenetsorszám> 1 és a STAT válaszában megadott üzenetszám között van, határokat is beleértve
 - Válasz:
 - -ERR No such message
 - +OK
Üzenet tartalma
Üzenet tartalma
.
- <- itt ebben a sorban egy pont (.) karakter található
- Azaz hiba, ha nincs ilyen üzenet
 - Ha van, akkor +OK utáni sorokban az üzenet tartalma, az utolsó sorban egy darab pont.
 - Az üzenet egy teljes, fejlécekkel ellátott e-mail üzenet
- Kilépés:
 - Kérés: QUIT
 - Válasz: +OK Bye

(Egy kis érdekesség: ha sikerült megoldani a feladatot, akkor a helyesen megírt kliensed használható tetszőleges, titkosítást nem használó POP3 szerveren. Ezek közül néhány példa: freemail.hu, pop3.citromail.hu). Aki kéri a ZH-t követően megkaphatja a kódját.

FIGYELEM, ha esetleg valamiért a felhasználónév-neptun nem működik a hitelesítésnél, akkor próbáld a „pelda”, „neptun” párost.