

Java programozás – 2. házi feladat

SpaceX űrhajó dokkolás szimulálása az ISS-re (szálkezelés)(100 pont)

A nemzetközi űrállomás továbbfejlesztett változata 5 db dokkoló állomással rendelkezik. Egy dokkoló állomást egyszerre csak egy űrhajó vehet igénybe. A SpaceX nevű cég turisták utaztatását végzi az űrállomásra. A feladat egy szimuláció elkészítése, ahol minden egyes részletet naplóz a központi számítógép.

ISS

- 5 db dokkoló
- Kapacitása (ennyi személy lehet egyszerre rajta): 250
- Max teherbírás: 2500

Űrhajó

- Minden űrhajó rendelkezik egy egyedi azonosítóval
- Minden űrhajónak van egy súlya (m , $100 < m < 1000$)
- Minden űrhajónak van egy kapacitása (c , $10 < c < 100$)
- 5-10 másodpercenként hozzunk létre egy űrhajót (az utasok számát és a súlyt véletlenszerűen generáljuk)
- Az űrhajó akkor dokkolhat az űrállomásra, ha van szabad dokkoló és elegendő hely van az utasok számára az ISS fedélzetén, továbbá a súlykorlátnak is megfelel
- Ha az űrhajó sikeresen dokkolt és minden utas kiszállt utána elindul vissza a Földre
minden űrhajó egy külön szál

Zsilipkapu

- Amikor egy űrhajó dokkolt az utasoknak egyenként át kell haladniuk egy zsilipkapun. (Minden dokkolónak saját kapuja van.) Egy utas áthaladása 0.5 másodpercet vesz igénybe.

Utas

- Minden utas rendelkezik egy egyedi azonosítóval
- Minden utas külön szál
- Minden utas véletlenszerű ideig tartózkodik az ISS fedélzetén (t [sec], $20 < t < 60$)
- Amikor letelt az idejük "jelentkeznek", hogy vissza szeretnének térni a Földre, ha elegendő utas gyűlik össze ($50 <$), a Földről érkezik egy űrhajó a megfelelő kapacitással és egy véletlen súllyal, (rá is vonatkoznak a dokkolási szabályok)
- A visszatérő utasok szintén áthaladnak egyenként a zsilipkapun, majd amikor az űrhajó megtelt, elindul vissza a Földre

Központi számítógép

- Feljegyez minden eseményt és naplózza egy fájlba, pl.:
 - o Űrhajó 1 elindult az ISS re, 100 utassal, 1000 súllyal
 - o Űrhajó 2 elindult az ISS re, 34 utassal, 543 súllyal
 - o Űrhajó 1 dokkol a 2. dokkolóhoz
 - o Az ISS új súlya és kapacitása
 - o Utas 1 áthaladt a zsilipkapun
 - o Utas 2 áthaladt a zsilipkapun
 - o Utas 100 vissza kíván térni a Földre
 - o Űrhajó 10 várakozik a dokkoláshoz
 - o ...
- Az űrhajók a központi számítógépen keresztül ellenőrzik, hogy az ISS en van e hely és a súly is megfelelő e a dokkoláshoz
- A központi számítógép összesíti a hazautazási kérelmeket és ha elég számú gyűlik össze, akkor indít egy üres űrhajót
- A központi számítógép egy (vagy több) külön szál
- Ha túl sok űrhajó várakozik a dokkoláshoz ($10 <$), a központi számítógép szünetelteti az újabb űrhajók felküldését, kivéve az üres űrhajókét. (Ha túl sok ember van az ISS en akkor nem tud teli űrhajó dokkolni, de üres igen)

Amikor létrejön egy űrhajó (egy új szál), akkor azonnal létrejönnek az utasokat reprezentáló szálak is és elindulnak. (Amíg az utasok az űrhajón vannak addig várakoznak.) Tekintsük úgy, hogy az űrhajó már az ISS körül jön létre, tehát nem kell foglalkozni a Föld-ISS távolság megtételével. Amikor egy űrhajó (utasokkal, vagy azok nélkül) elhagyja az űrállomást még 10 mp-ig fut a szál majd megszűnik mind az űrhajó, mind a benne lévő utas szálak.

Optimalizációs részfeladat

Ha az optimalizáció be van kapcsolva (egy bool flag), akkor az üres űrhajó nem dokkol ki azonnal, hanem a központi számítógép ellenőrzi, hogy mi lenne az előnyösebb:

- Maradjon bent az üres űrhajó, mert éppen elég jelentkező van a visszatérésre a Földre, vagy
- Mégis dokkoljon ki és álljon be egy teli űrhajó
- Minél több utas van a fedélzeten annál előnyösebb (több fizető vendég), viszont ha egy űrhajó üresen tér vissza a Földre az pazarlás, tehát el kell döntenie, hogy az éri e meg jobban, ha egy teli űrhajó bedokkol és pár utas feljut az ISS-re, vagy esetleg már túl sok a vissza kíváncozó utas, így előnyösebb egy bent lévő űrhajónak megvárni őket.

Leadási határidő: 2019. április 11. 23:59.