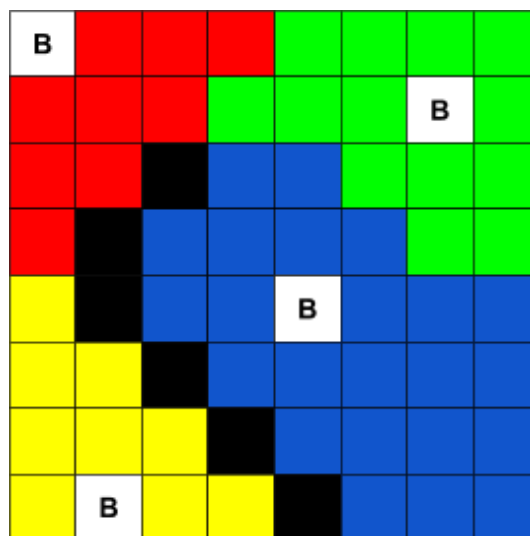


Observer pattern implementációja, szálakkal

A mobil szolgáltatók egy adott területet cella szinten osztanak fel, mely cellák alapján lehet lokalizálni az adott készülékeket. A feladatban adott egy $N \times M$ dimenziójú rács, mely cellái tartalmazzák, hogy melyik bázisállomáshoz tartoznak.

A szolgáltatók minden egyes cellára kiszámolják, hogy melyik bázisállomáshoz (B) tartozik az adott cella ld. 1. Ábra.

- 1. feladat:** Minden cellára határozd meg, hogy melyik bázisállomáshoz tartozik. Két cella között a távolságot euklideszi távolságként értelmezzük, amit a cellák (sor, oszlop) pozícióiból határozhatunk meg. Azokat a cellákat, melyek több bázisállomástól is ugyanolyan távol vannak (tehát nincs egyértelműen legközelebbi) véletlen módon rendeljük az egyik bázisállomáshoz.

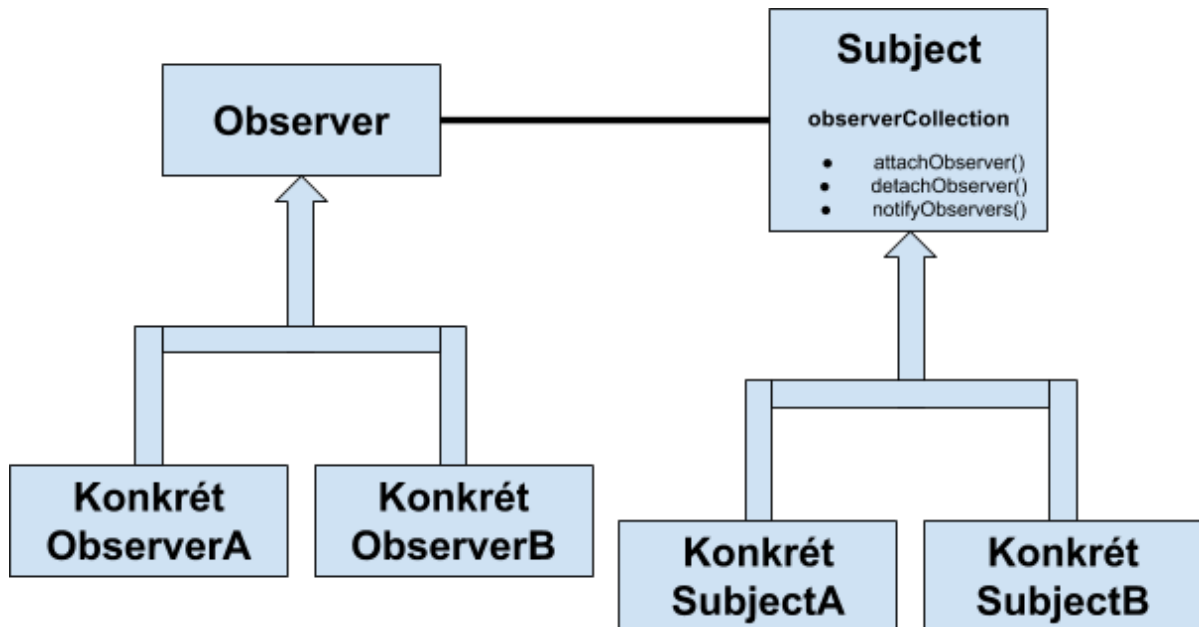


Az adott rácson belül mobil készülékek helyezkednek el, melyek másodpercenként véletlenszerűen változtatják a pozíciójukat.

- 2. feladat:** Hozz létre 10 mobil készüléket és helyezd el őket a rács egy tetszőleges cellájába. Rendelj a mobil készülékekhez új véletlen cella pozíciót minden másodperc letelte után. (Vigyázz, hogy a pozíció csak a megadott területen belül lehet.)

A kettes feladattal szimuláljuk a készülékek mozgását. Minden készülék kapcsolódik a hozzá legközelebbi bázisállomáshoz, amit tudunk a cella információjából, hiszen az egyes feladatban kiszámoltuk, melyik cella melyik bázisállomáshoz tartozik. Ahogy a mobil készülékek változtatják a pozíciójukat úgy frissül a legközelebbi bázisállomás kiléte.

- 3. feladat:** Az *Observer* minta implementálásával valósítsuk meg a bázisállomások és a mobil készülékek kommunikációját.



1. A mobilok minden másodpercben megváltoztatják a pozíciójukat (ld. 2. feladat). A mobil iratkozzon le a régi bázisállomásról és iratkozzon fel az új állomásra, melyet a cellák határoznak meg (ld. 1. feladat). A bázisállomások minden másodpercben elküldik az összes rájuk feliratkozott Mobilnak az adott állomás nevét. A mobilok az update() metódusok által fogadják az üzenetet és kiírnak a konzolra egy **id-t** mely azonosítja a mobilt és kiírja a kapott üzenetet, azaz a küldő állomás nevét.
2. A mobilok 20% valószínűséggel kérnek a saját bázis állomásuktól egy dátum frissítést. A bázis állomás a kérésre elküldi a frissítést az adott készüléknek, mely kiírja azt a konzolra.

Példa kimenet:

```

Mon Mar 13 15:02:13 CET 2017
Mon Mar 13 15:02:13 CET 2017
Mobile 7 get the following message: Station_8
Mobile 2 get the following message: Station_8
Mobile 9 get the following message: Station_8
Mobile 1 get the following message: Station_0
Mobile 3 get the following message: Station_1
Mobile 0 get the following message: Station_1
Mobile 8 get the following message: Station_1
Mobile 6 get the following message: Station_1
Mobile 4 get the following message: Station_2
Mon Mar 13 15:02:14 CET 2017
Mobile 5 get the following message: Station_2
Mon Mar 13 15:02:14 CET 2017
Mon Mar 13 15:02:14 CET 2017
Mobile 7 get the following message: Station_9
Mobile 9 get the following message: Station_1
  
```

A kiadott kód tartalmazza az implementálandó osztályokat és a main-t. A main csak egy vázlat, amit megfelelően változtatni kell az osztályok implementációjának függvényében. Minden mobil és minden bázisállomás példánynak külön szálon kell futnia!