## 6. laboratorijska vježba

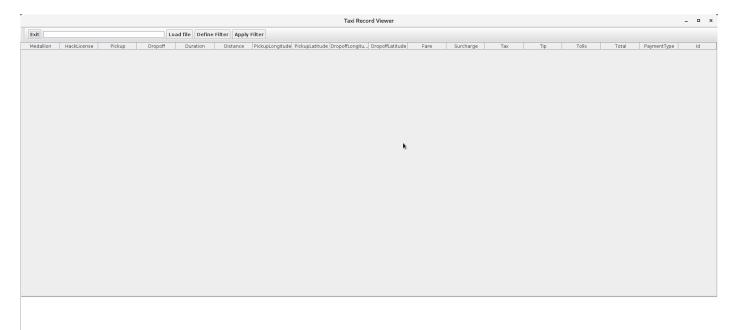
Važna napomena: u svim zadacima potrebno je napisati Javadoc komentare za svaki razred te generirati dokumentaciju. Svi nazivi razreda, metoda i varijabli moraju biti na engleskom. Sav napisani programski kod mora biti napisan u skladu s konvencijama imenovanja varijabli, metoda i razreda (varijable i metode: malo početno slovo, camel-case; razredi i sučelja: veliko početno slovo, camel-case; konstante: uobičajeno sve veliko i razdvajanje podvlakom) te ostalim pozitivnim praksama (uključivo i korektno uvlačenje redaka; smisleno razdvajanje više različitih semantički grupiranih redaka praznim redcima, pravilnim razmještajem zatvorene vitičaste zagrade i slično). Za više informacija pogledajte http://www.oracle.com/technetwork/java/codeconventions-150003.pdf

# Zadatak 1 - Filtriranje zapisa

Potrebno je napraviti aplikaciju za prikaz i filtriranje podataka o vožnjama taksija koji su bili korišteni na natjecanju <u>ACM DEBS 2015 Grand Challenge</u>. Ovi podaci su veličine 18,6 MB i mogu skinuti sa sljedeće poveznice: <a href="https://www.dropbox.com/s/2ymrt3vnz1cx1tn/data\_small.csv?dl=0">https://www.dropbox.com/s/2ymrt3vnz1cx1tn/data\_small.csv?dl=0</a>. Podaci u ovoj datoteci su u tekstualnom (CSV) obliku gdje svaka linija predstavlja zapis o jednoj taksi vožnji. Linija se sastoji od vrijednosti (medallion, hack\_license, pickup\_datetime, dropoff\_datetime ...) odvojenih zarezom kao što je objašnjeno na web-stranici natjecanja <u>ACM DEBS 2015 Grand Challenge</u>.

Podaci o vožnjama taksija trebaju biti prikazani u tablici. Kako Swingova tablica (JTable) nije obrađena na predavanjima, za prikaz podataka ćemo koristiti unaprijed pripremljenu klasu GenericTablePanel koja omogućava tablični prikaz liste objekata nekog tipa (pod uvjetom da objekti tog tipa imaju barem jedan javni getter) u tabličnom obliku. Ova klasa i primjer njenog korištenja se nalaze na sljedećoj poveznici: https://github.com/FER-OOP/Lectures/tree/master/Lab6/GenericTable.

Aplikacija treba izgledati kako je prikazano na sljedećoj slici (pri čemu redoslijed stupaca nije bitan).



Na vrhu se nalazi traka s alatima (JToolBar) koja ima 4 gumba i jedno tekstualno polje. Gumb Exit je namijenjen izlasku iz aplikacije. Gumb Load je namijenjen učitavanju datoteke s podacima o taksi vožnjama. Pri tome put do datoteke treba unijeti u tekstualno polje s lijeve strane ovog gumba. Učitavanje datoteke sa zapisima je potrebno ostvariti korištenjem statičke metode Stream<String> lines (Path path)

klase Files, tj. korištenjem kolekcijskih tokova na način da je iz toka String-ova potrebno stvoriti tok TaxiRecord-a kojeg treba terminirati u listu pa prikazati u tablici. Klikom na gumb Define Filter treba se otvoriti novi prozor unutar kojega se mogu definirati uvjeti na osnovu kojih će se filtrirati zapisi koji su trenutno prikazani u tablici. Ovaj prozor se ne može zatvoriti, već se samo može sakriti i to ili klikom na gumb x u njegovom gornjem desnom uglu ili ponovnim klikom na gumb Define Filter. Klikom na gumb Apply Filter je potrebno filtrirati zapise koji su trenutno prikazani u tablici primjenom definiranog filtra, nakon čega je potrebno u tablici prikazati samo one zapise koji su zadovoljili definirane uvjete. Filtriranje je potrebno ostvariti korištenjem kolekcijskih tokova tako da je iz liste podataka u tablici (oni se dohvaćaju pozivom metode getRecords kako je pokazano u primjeru korištenja klase GenericTablePanel) potrebno stvoriti tok te ga filtrirati na osnovu definiranih uvjeta. Prozor u kojem se definiraju uvjeti za filtriranje treba izgledati kako je prikazano na sljedećoj slici.



Mogu se definirati 4 uvjeta filtriranja: preskakanje prvih n zapisa, izostavljanje svih osim prvih n zapisa, izostavljanje zapisa čija i plaćanja ne odgovara odabranom i izostavljanje zapisa čija je vrijednost atributa distance manja ili "veća od ili jednaka" definiranoj. Za svaki uvjet postoji kvačica kojem se on može odabrati ili ne. Prozor s uvjetima za filtriranje je potrebno ostvariti u zasebnoj klasi koja nasljeđuje klasu JPanel ili JFrame i ima jednu metodu kojom se mogu dohvatiti odabrani uvjeti filtriranja. Uvjete filtriranja je potrebno ostvariti u zasebnoj klasi koja se sastoji od niza gettera i settera, poput klase UserData koja je pokazana na predavanjima.

U sredini aplikacije se nalazi tablica u kojoj se prikazuju zapisi, a na dnu se nalazi tekstualno područje (JTextArea) unutar kojeg se ispisuje log (sa porukama za korisnika) kao što je prikazano na sljedećoj slici. U logu treba ispisivati sve pogreške prilikom korištenja aplikacije (npr. nepostojanje datoteke pri učitavanju).

								Taxi Re	cord Viewer								-
Exit /kpripuzi	tt   /kpriptziic/Downloads/sorted_data.csv/data_small.csv   Load file   Define Filter   Apply Filter																
Medallion	HackLicense	Pickup	Dropoff	Duration	Distance	PickupLongitu	de PickupLatitu	de DropoffLongitu	DropoffLatitu	de Fare	Surcharge	Tax	Tip	Tolls	Total	PaymentTyp	е
4129883A1 7	7077F9FD5AD	Tue Jan 01 00:	Tue Jan 01 00:	120	0.61	-73.987373	40.724861	-73.983772	40.730995	4.0	0.5	0.5	0.0	0.0	5.0	CRD	4
A7F69213 0	0B7691D86D	Tue Jan 01 00:	Tue Jan 01 00:	60	0.0	0.0	0.0	0.0	0.0	2.5	0.5	0.5	0.25	0.0	3.75	CRD	5
BFA82ECA8 C	F8604E72D8	Tue Jan 01 00:	Tue Jan 01 00:	60	0.39	-73.981544	40.781475	-73.979439	40.784386	3.0	0.5	0.5	0.7	0.0	4.7	CRD	6
78CC6D4EC B	37567F5BFD5	Tue Jan 01 00:	Tue Jan 01 00:	240	1.21	-73.973	40.79314	-73.981453	40.778465	6.0	0.5	0.5	1.3	0.0	8.3	CRD	9
29E9A69B E	D368552102	Tue Jan 01 00:	Tue Jan 01 00:	180	0.74	-73.971138	40.75898	-73.972206	40.752502	4.5	0.5	0.5	0.0	0.0	5.5	CRD	10
90377AB7 0	00B7691D86D	Tue Jan 01 00:	Tue Jan 01 00:	60	0.0	0.0	0.0	0.0	0.0	3.0	0.5	0.5	0.07	0.0	4.07	CRD	11
0B52B22B 0	0B7691D86D	Tue Jan 01 00:	Tue Jan 01 00:	60	0.0	-73.937637	40.758369	-73.937607	40.75835	2.5	0.5	0.5	0.25	0.0	3.75	CRD	15
E773C92F B	36EAE07E2AD	Tue Jan 01 00:	Tue Jan 01 00:	180	1.55	-74.003197	40.733032	-74.012985	40.717377	6.0	0.5	0.5	1.62	0.0	8.62	CRD	18
9D2C8C830 A	A2AB59B1DBA	Tue lan 01 00:	Tue Jan 01 00:	240	0.59	-74.004517	40.721241	-73.997459	40.719055	5.0	0.5	0.5	1.1	0.0	7.1	CRD	21
0493FEB9 0	00B7691D86D	Tue Jan 01 00:	Tue Jan 01 00:	120	0.0	-73.802734	40.705162	-73.802734	40.705162	3.0	0.5	0.5	0.45	0.0	4.45	CRD	23
BDF2DC5F1 4	BF10A3C62F	Tue Ian 01 00:	Tue Ian 01 00:	201	0.5	-73.989891	40,739761	-73,994003	40.73444	4.5	0.5	0.5	1.1	0.0	6.6	CRD	29
7DD250F4 4	E0CF2EFC2B	Tue Jan 01 00:	Tue Jan 01 00:	420	1.22	-73.996132	40.739567	-74.004288	40.750999	7.0	0.5	0.5	2.0	0.0	10.0	CRD	31
		Tue Jan 01 00:			0.91	-73.992805	40,73407	-73.981659	40,736805	5.5	0.5	0.5	1.2	0.0	7.7	CRD	33
		Tue lan 01 00:			0.61	-73.994644	40.756016	-73.998413	40.750599	4.0	0.5	0.5	1.35	0.0	6.35	CRD	36
BE818E3C 4	D85F9862CD	Tue Jan 01 00:	Tue Ian 01 00:	420	1.49	-73.985603	40.744263	-73.969627	40.757374	7.0	0.5	0.5	1.88	0.0	9.88	CRD	38
		Tue Jan 01 00:			1.0	-73.970779	40.798622	-73.979286	40.786877	5.5	0.5	0.5	1.2	0.0	7.7	CRD	40
		Tue Jan 01 00:			1.54	-73.977806	40.788998	-73.967438	40.806087	7.0	0.5	0.5	1.5	0.0	9.5	CRD	44
		Tue Jan 01 00:			0.84	-73.953842	40.778934	-73.954048	40.787163	5.0	0.5	0.5	1.0	0.0	7.0	CRD	54
		Tue Jan 01 00:			0.47	-73.981209	40.744404	-73.97715	40.745323	4.5	0.5	0.5	1.0	0.0	6.5	CRD	56
		Tue Jan 01 00:			2.73	-73,973907	40.747868	-73,948898	40.776649	9.5	0.5	0.5	2.0	0.0	12.5	CRD	59
		Tue Jan 01 00:			1.16	-74,003708	40.731953	-73,991974	40.735298	6.0	0.5	0.5	1.3	0.0	8.3	CRD	62
		Tue Jan 01 00:			2.26	-73.989853	40.738632	-73.966682	40.753292	8.5	0.5	0.5	1.8	0.0	11.3	CRD	63
		Tue Jan 01 00:			1.66	-73.977051	40.743141	-73,985184	40.723808	8.0	0.5	0.5	1.5	0.0	10.5	CRD	65
		Tue Jan 01 00:			1.1	-73.997749	40.756626	-73.988983	40.768436	5.0	0.5	0.5	1.0	0.0	7.0	CRD	66
		Tue Jan 01 00:			1.68	-73.981575	40.767632	-73.977737	40.757927	8.0	0.5	0.5	1.7	0.0	10.7	CRD	68
		Tue Jan 01 00:			1.67	-73.955521	40.779541	-73,964317	40.792328	7.5	0.5	0.5	3.3	0.0	11.8	CRD	72
		Tue Jan 01 00:			2.14	-73.975723	40.781693	-73.969147	40.760975	9.5	0.5	0.5	3.33	0.0	13.83	CRD	81
		. Tue Jan 01 00:			1.94	-73,975159	40.741474	-73.975021	40.720013	7.5	0.5	0.5	0.0	0.0	8.5	CRD	95
		Tue Jan 01 00:			1.41	-73.970123	40.784279	-73.954765	40.773224	7.0	0.5	0.5	1.0	0.0	9.0	CRD	96
		Tue Jan 01 00:			1.02	-73.990761	40.736568	-73.99015	40.746487	6.0	0.5	0.5	3.0	0.0	10.0	CRD	97
		Tue Jan 01 00:			1.18	-73.961761	40.715897	-73.957977	40.708416	6.5	0.5	0.5	1.4	0.0	8.9	CRD	101
		Tue Jan 01 00:			0.0	-73.937752	40.758381	-73.937737	40.758579	4.5	0.5	0.5	1.25	0.0	6.75	CRD	103
		Tue Jan 01 00:			0.0	-73.937744	40.758354	-73.937752	40.758358	4.0	0.5	0.5	0.15	0.0	5.15	CRD	104
		Tue Jan 01 00:			0.57	-73.963844	40.761364	-73,969093	40.754208	4.0	0.5	0.5	0.13	0.0	5.9	CRD	105
		Tue Jan 01 00:			1.88	-73.956558	40.771172	-73.975258	40.752247	7.5	0.5	0.5	1.6	0.0	10.1	CRD	107
		Tue Jan 01 00:			0.75	-73.94252	40.790524	-73.940559	40.79734	4.0	0.5	0.5	0.9	0.0	5.9	CRD	108
		Tue Jan 01 00:			2.77	-73.989616	40.729988	-74.013779	40.705036	10.5	0.5	0.5	2.2	0.0	13.7	CRD	115
		Tue Jan 01 00:			1.36	-73.969093	40.729988	-73,970131	40.690693	6.5	0.5	0.5	1.75	0.0	9.25	CRD	128
		Tue Jan 01 00:			1.42	-73.955872	40.768158	-73.947624	40.784519	6.0	0.5	0.5	3.0	0.0	10.0	CRD	128
		Tue Jan 01 00:			0.69	-73.996445	40.725719	-73.993294	40.719318	5.0	0.5	0.5	1.1	0.0	7.1	CRD	131
		Tue Jan 01 00:			0.82	-73.996445	40.725719	-73.993294	40.719318	7.5	0.5	0.5	1.6	0.0	10.1	CRD	131
		Tue Jan 01 00:			1.04	-74.001602	40.747089	-74.010002	40.744083	6.0	0.5	0.5	1.0	0.0	8.0	CRD	134
		Tue Jan 01 00:			1.04	-74.001602	40.721382	-74.010002	40.709789	0.0	0.5	0.5	1.0	0.0	8.0	CRD	134

Zbog složenosti posla je učitavanje i filtriranje zapisa potrebno ostvariti korištenjem klase SwingWorker, a ne korištenjem grafičke dretve. Pri tome je pri učitavanju potrebno onemogućiti gumb za primjenu filtra i obratno. Osim toga je svakom zapisu prilikom učitavanja potrebno pridodati redni broj (krajnji lijevi stupac na slici iznad). Ovaj broj se ne nalazi u tekstualnoj datoteci iz koje se učitavaju zapisi, već je pridodan prilikom učitavanja zapisa.

Primijetite da se klikom na ime stupca klasa GenericTablePanel omogućava sortiranje zapisa po tom stupcu.

## Zadatak 2 – Igra života

U ovom zadatku treba napraviti igru života (Conway's Game of Life - <a href="https://en.wikipedia.org/wiki/Conway's\_Game\_of\_Life">https://en.wikipedia.org/wiki/Conway's\_Game\_of\_Life</a>). Igra se sastoji od ploče koja je preko rubova spojena lijevo-desno i gore-dolje. Svako polje može imati dva stanja živo i mrtvo. Jedna iteracija se izvršava tako da se za svako polje ispita da li će u sljedećoj iteraciji biti živo ili mrtvo. Za izračunavanje se koriste sljedeća pravila:

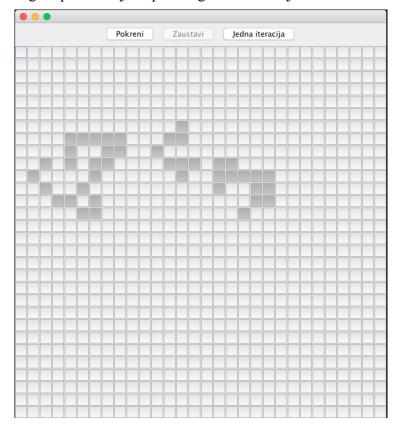
- Ako živo polje ima manje od dva živa susjedna polja umire (u sljedećoj je iteraciji mrtvo).
- Ako živo polje ima dva ili tri susjedna polja u polje ostaje živo.
- Ako živo polje ima više od tri živa susjeda umire.
- Mrtvo polje koje ima točno tri živa susjeda postaje živo (rađa se).

Svako polje ima 8 susjeda, a to su polja oko tog polja horizontalno, vertikalno i dijagonalno.

Igru života potrebno napraviti u klasi Board. Klasa ima konstruktor koji prima veličinu polja (npr. 30, 30). Metoda boolean isCellAlive(int x, int y) vraća istinu ako je polje s koordinatama živo true, a inače false. Metode getWidth() i getHeight() vraćaju veličinu polja. Metoda setCell(int x, int y, boolean alive) postavlja stanje jednog polja. Metoda int countAliveNeighbors(int x, int y) računa žive susjede. Metoda playOneIteration() prolazi kroz sva polja, izračunava nove vrijednosti i nakon njenog izvršavanja ploča ima vrijednosti nakon jedne iteracije. Metoda addListener (BoardListener listener) dodaje slušača u listu, a removeListener (BoardListener listener) ga miče. Nakon što se izvrši jedna iteracija sve slušače se obavještava o promjeni na ploči tako da se pozove metoda iz sučelja BoardListener. Sučelje izgleda ovako:

```
public interface BoardListener {
    void boardChanged(Board board);
}
```

Nakon što je napravljena logika potrebno je napraviti grafičko sučelje kao na slici.



#### Objektno orijentirano programiranje (ak.god. 2018./2019.) – 6. laboratorijska vježba

Grafičko sučelje je napravljeno u klasi BoardFrame. BoardFrame inicijalizira ploču od 30x30 polja. Svako polje je prikazano pomoću JToggleButton-a. Kada se klikne na pojedini gumb onda se na ploči treba podesiti vrijednost polja ovisno o tome je li gumb ostao pritisnut ili je otpušten.

### Na vrhu su tri gumba:

- «Pokreni» pokreće instancu klase SwingWorker koja izračunava novu iteraciju svakih 500ms i
  osvježava polja u grafičkom sučelju. Osvježavanje se mora izvršavati u grafičkoj (event dispatch)
  dretvi. Kada se klikne na pokreni onda se taj gumb onemogućava, ali omogući klik na gumb zaustavi.
- «Zaustavi» zaustavlja izvršavanje *workera* koja se pokreće pritiskom na pokreni. Nakon toga se onemogućava gumb zaustavi, ali se omogući gumb pokreni.
- «Jedna iteracija» ovaj gumb je omogućen samo kada nije pokrenut worker koja izračunava. Ako je omogućen i kliknuto je na njega onda se pokreće izvršavanje jedne iteracije. Izvršavanje se može izvršiti u grafičkoj dretvi.