

Praktikum iz objektno-orijentisanog programiranja (13S112POOP) Projektni zadatak – Java

Napisati skup klasa za realizaciju softverskog sistema za vizuelizaciju grafovskih fajlova. Podržani formati grafovskih fajlova treba da budu GML i CSV, a treba predvideti i mogućnost proširenja drugim formatima. Opis formata navedenih fajlova je dat u prilogu ovog dokumenta.

Korisnik (naručilac) softvera, želi da softver pruži sledeće funkcionalnosti:

- Interakciju sa korisnikom putem grafičkog korisničkog interfejsa
- Učitavanje podataka o grafu iz ulaznih fajlova
- Prikazivanje grafa i osnovnih informacija uz sledeće mogućnosti
 - Zoom in/zoom out
 - Pomeranje prikazanog grafa ili pojedinačnih čvorova
 - Expansion/Contraction i Force Atlas algoritmi
 - Menjanje boje i veličine čvorova, grana i labela
 - Formatiranje grafa (čvorova ili labela) na osnovu stepena čvora
- Osnovnu manipulaciju nad grafom
 - Selekcija čvorova i grana
 - Brisanje, dodavanje i izmena čvorova i grana
 - Određivanje najkraćeg puta između čvorova
- Dodatne funkcionalnosti
 - Eksportovanje i importovanje formatiranog grafa
 - Undo/Redo opcija
- Kraj rada

Za uspešno rešenje zadatka potrebno je izvršiti analizu zahteva. Kao rezultat analize, potrebno je dopuniti i precizirati funkcionalnu specifikaciju softverskog alata. Na osnovu specifikacije, potrebno je napisati sistem klasa u jeziku Java koje realizuju traženi softver. U nastavku su navedeni neki elementi specifikacije. Od studenata se očekuje da dopune one stavke koje nisu dovoljno precizno formulisane, odnosno dodaju nove stavke (tamo gde to ima smisla) ukoliko uoče prostor za unapređenje. Izmene i dopune specifikacije mogu da donekle odudaraju od zahteva naručioca softvera u onoj meri u kojoj to neće narušiti traženu funkcionalnost. Prilikom izrade specifikacije voditi računa o potencijalnom unapređenju softvera na osnovu naknadnih zahteva.

Prilikom izrade rešenja, od studenata se očekuje intenzivno korišćenje svih onih mogućnosti koje pruža specifikacija jezika Java, kao što su kolekcije, algoritmi, regularni izrazi, iteratori, lambda izrazi i sl. **Rešenja koja ne vode računa o ovom aspektu neće moći da dobiju maksimalan broj poena.** Takođe, voditi računa o objektno orijentisanom dizajnu rešenja, čistoći, čitkosti i komentarisanju programskog koda.

Funkcionalna specifikacija

U nastavku je zadat deo korisničkih zahteva koje treba razraditi i, po potrebi, dopuniti tako da se dobije funkcionalna aplikacija.

Interakcija sa korisnikom

Korisnik može da interaguje sa programom izborom u datom trenutku dostupnih opcija putem grafičkog korisničkog interfejsa. Obezbediti da se interakcija sa korisnikom odvija putem statičkih ekrana. Interakcija može da se vrši putem tastature ili miša. U zavisnosti od izabrane opcije i njenih parametara, program izvršava zadatu opciju ili ispisuje poruku greške. Poruka greške treba da bude što je moguće detaljnija da bi korisniku pomogla da grešku otkloni. Sve eventualne parametre koji su potrebni prilikom rada aplikacije je potrebno zatražiti od korisnika. Ukoliko korisnik ne zada ništa, koristiti vrednosti fiksirane u programu.

Učitavanje, prikazivanje grafa i osnovna manipulacija

Potrebno je obezbediti učitavanje podataka o grafu iz fajlova navedenih formata čije su specifikacije date na kraju ovog dokumenta. Po učitavanju podataka potrebno je učitan graf prikazati korisniku zajedno sa osnovnim informacijama o grafu (broj čvorova, broj grana). Početni prikaz grafa koristi nasumičan raspored čvorova uz izbegavanje preklapanja čvorova ukoliko je to moguće. Po učitavanju grafa potrebno je korisniku obezbediti sledeće mogućnosti:

- Selekcija, brisanje, izmena i dodavanje čvorova i grana grafa
- Računanje i prikaz najkraćeg puta između dva selektovana čvora grafa
- Zoom in/Zoom out – uvećavanje i umanjivanje trenutnog prikaza grafa
- Pomeranje celokupnog prikaza grafa ili pojedinačnih čvorova
- Menjanje boje i veličine selektovanih čvorova, labela i grana
- Formatiranje boje i veličine čvorova i labela na osnovu stepena čvora grafa (stepen predstavlja broj grana koje izlaze i ulaze u čvor)
- Uključivanje i isključivanje prikaza labela na čvorovima

Pored navedenih mogućnosti potrebno je podržati i Expansion/Contraction i Force Atlas algoritam. Rad navedenih algoritama je moguće pokrenuti i zaustaviti, a njihovu realizaciju je potrebno obaviti u posebnoj niti. Expansion/Contraction algoritmi skaliraju trenutni prikaz grafa na ekranu i za razliku od Zoom in/Zoom out opcija utiču na veličinu ekportovanog prikaza grafa. Algoritam Force Atlas koristi se za raspoređivanje čvorova grafa rada lakše vizuelizacije i interpretacije. Detaljno objašnjenje rada ovog algoritma moguće je naći na sledećoj web adresi: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4051631/>

Dodatne funkcionalnosti

Pored navedenih manipulacija vezanih za prikaz i formatiranje grafa korisniku je potrebno obezbediti i sledeće funkcionalnosti:

- Eksportovanje grafa u vidu slike
- Eksportovanje grafa u formatu koji je potrebno definisati i koji je kasnije moguće iskoristiti za učitavanje grafa. Format mora da obezbedi čuvanje svih informacija o grafu i formatiranju.
- Undo/Redo operacije – sve akcije vezane za izmenu grafa, prikaza i formatiranje moguće je poništiti opcijom Undo, a poništene akcije je moguće ponovo izvršiti opcijom Redo.

Kraj rada

Korisnik može da zahteva kraj rada. Od korisnika se traži potvrda za napuštanje programa, uz upozorenje ukoliko program nije sačuvao sve izmene grafa. Korisniku treba ponuditi mogućnost da sačuva izmene pre napuštanja programa, ukoliko to želi.

Specifikacija GML formata

GML (Graph Modeling Language) je tekstualni format fajla za predstavljanje grafova pomoću veoma jednostavne sintakse. Koriste ga alati Gephi, Graphlet, Pajek, zEd, LEDA i NetworkX.

Specifikacija grafa se sastoji od navođenja čvorova i grana grafa po format koji je prikazan u primeru, uz mogućnost zadavanja i labele za čvor ili granu.

Primeri fajlova u ovom formatu:

```
graph
[
  node
  [
    id A
  ]
  node
  [
    id B
  ]
  node
  [
    id C
  ]
  edge
  [
    source B
    target A
  ]
  edge
  [
    source C
    target A
  ]
]
```

```
graph
[
  node
  [
    id A
    label "Node A"
  ]
  node
  [
    id B
    label "Node B"
  ]
  node
  [
    id C
    label "Node C"
  ]
  edge
  [
    source B
    target A
    label "Edge B to A"
  ]
  edge
  [
    source C
    target A
    label "Edge C to A"
  ]
]
```

Specifikacija CSV formata

Alat Gephi podržava i CSV format pomoću koga je lako predstaviti grafove. CSV format podržavaju i baze podataka, kao i Microsoft Excel. Svaka linija kod ovog formata mora sadržati bar dva elementa odvojena specijalnim znakom – separatorom, za koji se uglavnom koristi zarez. Vrednosti se mogu pisati pod navodnicima u slučaju kada sadrže specijalni znak.

Predstavljanje grafova pomoću ovog formata moguće je realizovati na više načina. Prvi način je lista grana i kod ovog načina se u svakom redu zadaju dva identifikatora čvora, koji su povezani granom grafa, odvojena separatorom. U sledećem primeru predstavljen je graf koji ima grane između čvorova a-b i čvorova b-c.

```
a;b  
b;c
```

Drugi način za reprezentaciju grafa pomoću ovog formata jeste lista susednosti. U svakom redu navodi se identifikator čvora grafa i svih ostalih čvorova sa kojima je on povezan nekom granom grafa. U navedenom primeru predstavljen je graf koji ima grane između čvorova a-b, b-c i b-d.

```
a;b  
b;c;d
```

Treći način za predstavljanje grafa ovim formatom jeste kvadratna matrica u kojoj se pomoću jedinice označava postojanje grane između dva čvora, kao što je i prikazano u sledećem primeru:

```
;A;B;C;D;E  
A;0;1;0;1;0  
B;1;0;0;0;0  
C;0;0;1;0;0  
D;0;1;0;1;0  
E;0;0;0;0;0
```