

Obligatorisk oppgave 7: Labyrint GUI

INF1010

Frist: mandag 22. mai 2017 kl. 12:00

Versjon 1.0 (3bf3b0c)

Innhold

1 Innledning	1
2 Krav	1
3 Konvertere løsningstrengen fra oblig 5	2
4 Valgfri del	3
4.1 Bla i løsningene	3
4.2 Programmeringskonkurranse	3

1 Innledning

I denne oppgaven skal du bruke JavaFX for å lage et grafisk brukergrensesnitt (GUI) for labyrintprogrammet du lagde i [oblig 5](#).

2 Krav

Programmet skal bruke klassen `javafx.stage.FileChooser` for å la brukeren finne/velge filen med labyrinten.

Når brukeren har valgt labyrintfilen som skal åpnes, skal labyrinten vises grafisk. For den grafiske representasjonen av labyrinten skal programmet bruke klassen `javafx.scene.layout.GridPane`. Når brukeren klikker på en hvit rute, skal du bruke programmet fra oblig 5 for å finne alle løsningene fra denne ruten. Du skal ta én løsning fra beholderen med løsninger, og vise denne i labyrinten. Er det

flere enn én løsning, kan du vise den første, med informasjon om hvor mange løsninger som totalt ble funnet.

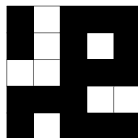
3 Konvertere løsningstrengen fra oblig 5

I programmet fra oblig 5 skulle hver løsning være gitt på et slikt format:

```
(2, 2) --> (2, 3) --> (1, 3)
```

Det kan være vanskelig å hente ut relevant informasjon fra denne strengen for å utheve de riktige rutene i grensesnittet. Du kan derfor bruke følgende metode for å konvertere en løsningstreng til et 2D-array med boolske verdier (`boolean[][]`). Verdiene som er `true` indikerer at ruten er en del av løsningveien. Verdiene som er `false` indikerer at ruten ikke er en del av løsningen.

```
/**
 * Konverterer losning-String fra oblig 5 til en boolean[][]-representasjon
 * av losningstien.
 * @param losningString String-representasjon av utveien
 * @param bredde         bredde til labyrinten
 * @param hoyde          hoyde til labyrinten
 * @return               2D-representasjon av rutene der true indikerer at
 *                       ruten er en del av utveien.
 */
static boolean[][] losningStringTilTabell(String losningString, int bredde, int hoyde) {
    boolean[][] losning = new boolean[hoyde][bredde];
    java.util.regex.Pattern p = java.util.regex.Pattern.compile("\\s*([0-9]+),([0-9]+)\\s*");
    java.util.regex.Matcher m = p.matcher(losningString.replaceAll("\\s", ""));
    while(m.find()) {
        int x = Integer.parseInt(m.group(1))-1;
        int y = Integer.parseInt(m.group(2))-1;
        losning[y][x] = true;
    }
    return losning;
}
```



Figur 1: Labyrinten i fil 7.

Et eksempelkall på `losningStringTilTabell` i fil 7 (fra testfilene til oblig 5) ser slik ut:

```
String losning = "(2, 2) --> (2, 3) --> (1, 3)"; // mulig utvei fra (2,2) i fil 7
losningStringTilTabell(losning, 5, 5);
```

Dette vil returnere et `boolean[][]`-objekt med følgende verdier (T representerer `true`, . representerer `false`):

```
. . . . .  
. T . . .  
T T . . .  
. . . . .  
. . . . .
```

4 Valgfri del

4.1 Bla i løsningene

Om det er flere løsninger, er det nyttig å kunne se alle løsningene til labyrinten. Utvid GUI-et med knapper slik at løsningene kan vises en etter en når brukeren trykker på en knapp. I tillegg til å vise antall løsninger kan det også være lurt å vise hvilken løsning som vises.

4.2 Programmeringskonkurranse

Tradisjonen tro arrangerer vi også i år en konkurranse i forbindelse med denne obligen. For mer informasjon om konkurransen, se [Konkurranse for oblig 7 på semestersidene](#).

Lykke til!

Stein Gjessing, Stein Michael og Mathias