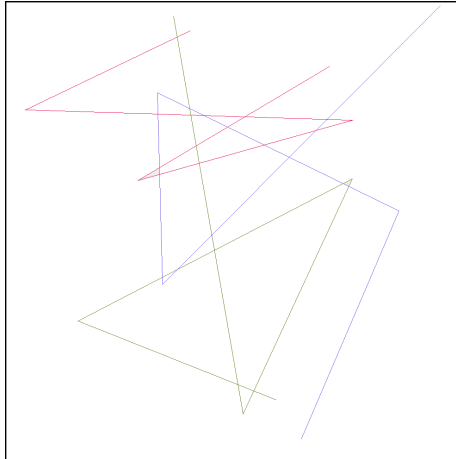


Seminarska naloga 2 - iskanje presečišč

Naloga

Napišite program, ki prešteje vsa presečišča podanih lomljenih črt z različnimi barvami. Rok za oddajo je 13.5.2018.

Figure 1: Primer treh lomljenih črt, ki tvorijo 18 presečišč.



Podrobnosti:

- Daljici, različne barve, ki se stikata se štejeta za presečišče.
- Daljici, ki se prekrivata, štejete kot eno presečišče.
- Več daljic se lahko seka v eni točki.
- Navpičnih daljic v testnih podatkih ni.
- Vse lomljene črte so enake dolžine in vsaka je različne barve.
- Vse daljice bodo znotraj kvadrata $[0,0]$ do $[1000, 1000]$.

Program mora teči v času $O(n \log(n))$, kjer je n število daljic oz. $O((n+k) \log(n))$, kjer je n število daljic in k število dobljenih presečišč. Algoritem za iskanje presečišč morate implementirati sami. Lahko uporabite že implementirane podatkovne strukture kot so iskalna drevesa, sezname, kopice, slovarji itd... Nalogo lahko rešite v Javi ali Pythonu.

Vhod

Vhod boste prejeli, kot ime vhodne *.txt datoteke. Vsaka vrstica predstavlja eno točko, ki je opisana s tremi vrednostmi x in y koordinato ter barvo. Točke enake barve, ki si sledijo ena drugi so povezane. Točke enake barve so vedno podane skupaj. Generator naključnih vhodov imate podan na učilnici.

Primer za pripadajočo sliko:

```
646.5821602909257, 951.3577109193919, 0
857.5884598068334, 458.2333050626706, 0
335.9524025416939, 203.8747819531316, 0
346.90742873967685, 617.314071997303, 0
946.8595742485053, 15.899959834469012, 0
406.5367203992265, 69.77557886640152, 1
49.57158909682802, 240.49265711418556, 1
758.8891282977459, 262.6712588023032, 1
293.1966764035244, 392.8300211005812, 1
707.0930113755271, 146.58670415052433, 1
370.8618354358446, 38.46122451826361, 2
520.5586068847992, 897.234285735065, 2
756.826923315635, 389.57083495431965, 2
163.90611322459637, 696.3900565931677, 2
591.052795035771, 866.8494992973393, 2
```

Ogrodje za Java. Prvi parameter v funkcijo main bo ime datoteke z vhodnimi podatki.

```
public class Seminar2{
    public Seminar2(){

        public static void main(String[] args){

        }
    }
}
```

Izhod

Na standardni izhod programa vrnete samo število presečišč.

Primer za pripadajočo sliko:

18

Poročilo

K vaši rešitvi oddajte tudi pisno poročilo. Poročilo naj vsebuje opis uporabljenega algoritma in opis izboljšav, če ste jih uporabili. Poročilo naj vsebuje tudi analizo časovne zahtevnosti vašega algoritma glede na število podanih daljic. K poročilu oddajte tudi vašo izvirno kodo.

Ocenjevanje

Ocena bo sestavljena iz pravilnosti algoritma (ali zazna prekrivajoče, stikajoče daljice in večkratna presečišča) in učinkovitosti algoritma (za koliko daljic še vedno dobi rešitev v času 10s).