

# Travelling Salesman Problem

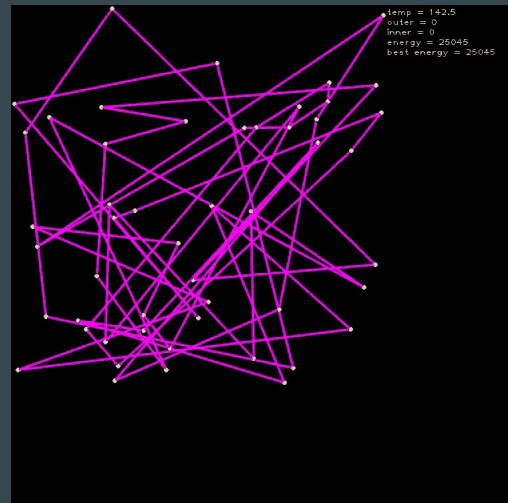
Predmetni Projekat iz Predmeta  
Operaciona Istraživanja

[github.com/stankovictab/ftn-oi-travelling-salesman](https://github.com/stankovictab/ftn-oi-travelling-salesman)

IN13-2018 Đorđe Stanković

# Travelling Salesman Problem

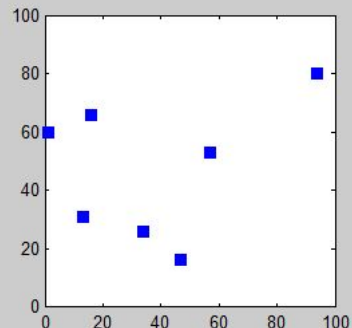
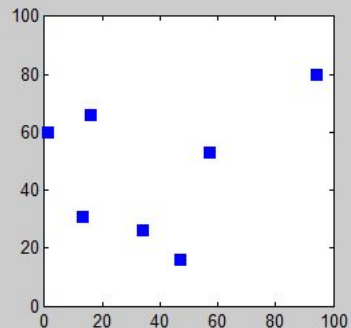
- Optimizacioni i kombinatorni problem
- Obilaženje svih  $n$  destinacija tačno jednom i vraćanje u početak
- Cilj je naći najpovoljniju putanju
- Reprezentacija preko grafa
- Težine grana mogu biti cena, distanca ili vreme
- Cene prelaza iz čvora u čvor se mogu predstaviti kvadratnom matricom cena prelaza



# Brute Force Algorithm

Brute Force algoritam uvek daje tačno rešenje, za razliku od ostalih algoritama (heuristika)

- Nalaženje svih mogućih permutacija
- Za svaku rutu naći ukupnu cenu
- Selektovati za optimalnu rutu onu sa najmanjom cenom
- Najsporiji
- $n!$  ruta



# Nearest Neighbour Algorithm

Nearest Neighbour algoritam nam daje rute formirane po najmanjim cenama prelaska u susede, za svaki čvor posebno.

Dakle, za  $n$  čvorova imamo  $n$  ruta.

- Naći najmanju cenu u redu matrice
- Preći u onaj red sa indeksom kolone selektovanog minimalnog elementa
- Ne sme se preći u čvor koji je već posećen - za ovo je u implementaciji korišćena "maska"

Nearest Neighbour je ujedno najbrži, ali i najjednostavniji.

Ne garantuje optimalnu rutu.

# Hungarian Algorithm

Mađarski Metod je algoritam za rešavanje problema dodele poslova. Svaki radnik može da radi bilo koji zadatak, ali sa različitom cenom. Cilj je pravilno raspodeliti radnike, sa najmanjom ukupnom cenom.

- Redukuje se početna matrica cena prelaza, po redu i koloni
- Radi se proveru jedinstvenih nula u redu ili koloni, i na odgovarajući način se puni "maska"
- Svaka selektovana nula predstavlja jedan selektovan prelaz
- Ako imamo tačno jednu selektovanu nulu u svakom redu i koloni, to znači da možemo da napravimo rutu

Može da se javi problem ako imamo 0 ili bar 2 nule u redu ili koloni u celoj redukovanoj matrici - to ukazuje na više optimalnih ruta.

# Genetic Algorithm

Genetski algoritam rešava optimizacione probleme po uzoru na proces prirodne selekcije.

- Inicijalizacija početne populacije na stohastičan način
- Svaka jedinka populacije je jedna ruta
- Određivanje prilagođenosti svake jedinke, odnosno ukupne cene rute
- Selekcija - Za roditelje se biraju rute sa najmanjom cenom
- Ukrštanje - Pravi se kombinacija čvorova dva roditelja, tako da se održi validnost rute
- Mutacija - Po određenoj verovatnoći će se neko dete mutirati
- Mutacija predstavlja rotaciju dva čvora u ruti

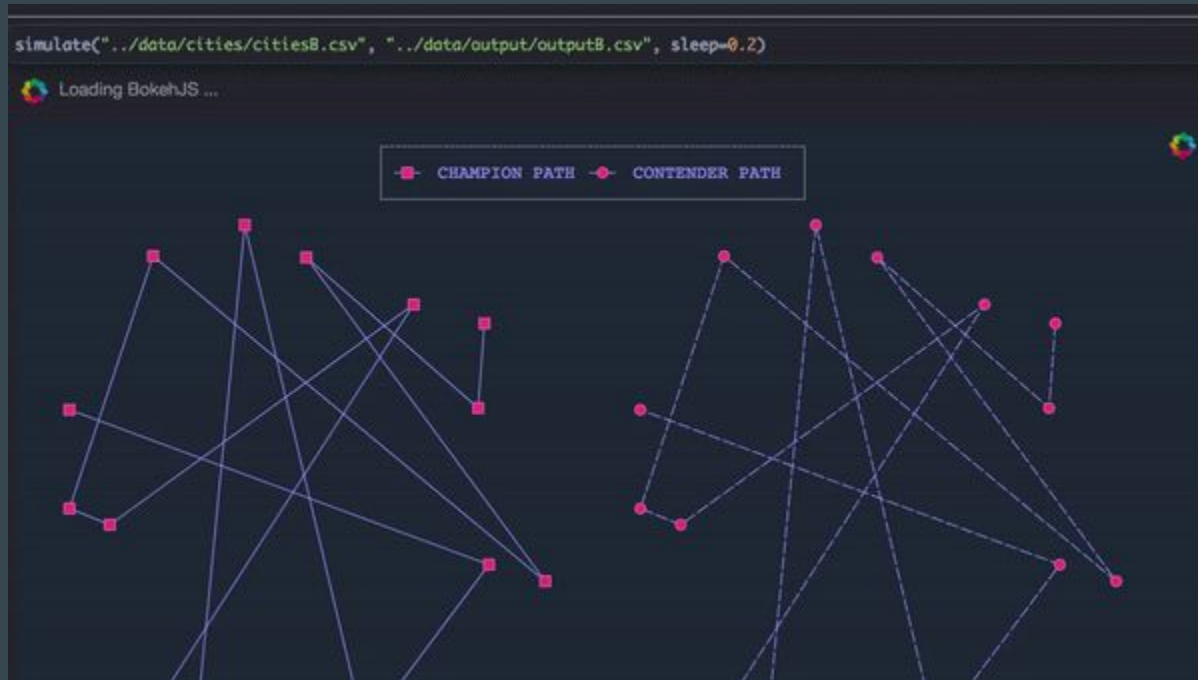
# Genetic Algorithm

- Proces generisanja novih populacija se ponavlja određen broj puta
- U poslednjoj populaciji za najbolju rutu bираmo onu sa najmanjom cenom
- Sa dobrim podešavanjima, poslednja populacija će konvergirati u minimalnu vrednost za optimalnu cenu rute

Ova implementacija ne pamti trenutnu najbolju rutu, već samo bira najbolju u poslednjoj populaciji.

Sa modifikacijama i takvim načinom pamćenja ruta, ovaj algoritam može da se pretvori u algoritam roja.

# Genetic Algorithm





# Hvala na Pažnji!

Source kod je dostupan na sledećem linku  
[github.com/stankovictab/ftn-oi-travelling-salesman](https://github.com/stankovictab/ftn-oi-travelling-salesman)

IN13-2018 Đorđe Stanković