

# Neighbour joining

Bioinformatika - projekt

*Filip Beć*

*Zorana Ćurković*

*Goran Gašić*

*Melita Kokot*

*Dino Šantl*

*Igor Smolković*

Fakultet elektrotehnike i računarstva

Zagreb, siječanj 2014.

# Uvod

- ▶ Izgradnja filogenetskog stabla
- ▶ Od dna prema vrhu
- ▶ Pohlepan algoritam - minimizira udaljenosti u svakom koraku
- ▶ Vremenska složenost  $O(N^3)$
- ▶ Memorijska složenost  $O(N^2)$

# Opis algoritma

1. Ulaz je matrica udaljenosti između genoma

# Opis algoritma

1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica -  $Q$

# Opis algoritma

1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica - Q
3. U matrici Q pronalazi se minimalna vrijednost između dva čvora

# Opis algoritma

1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica -  $Q$
3. U matrici  $Q$  pronalazi se minimalna vrijednost između dva čvora
4. Pronađena dva čvora spajaju se s novim čvorom

# Opis algoritma

1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica -  $Q$
3. U matrici  $Q$  pronalazi se minimalna vrijednost između dva čvora
4. Pronađena dva čvora spajaju se s novim čvorom
5. Iz matrice udaljenosti brišu se dva odabrana čvora i dodaje se novi čvor

# Opis algoritma

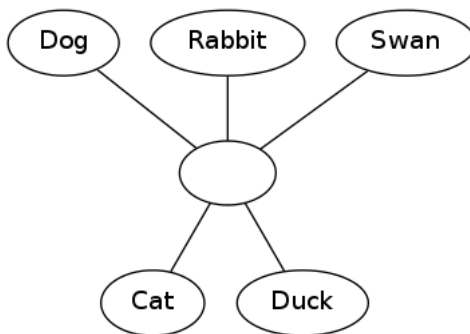
1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica - Q
3. U matrici Q pronalazi se minimalna vrijednost između dva čvora
4. Pronađena dva čvora spajaju se s novim čvorom
5. Iz matrice udaljenosti brišu se dva odabrana čvora i dodaje se novi čvor
6. Ako matrica udaljenosti sadrži više od tri čvora, skoči na 2. korak



# Opis algoritma

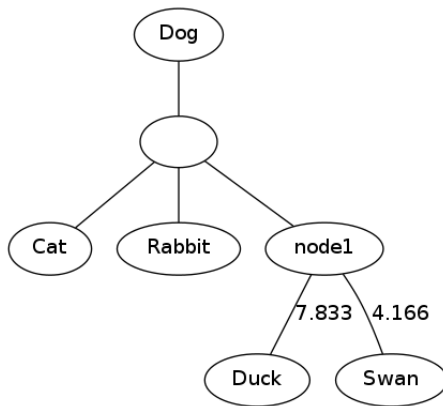
1. Ulaz je matrica udaljenosti između genoma
2. Iz matrice udaljenosti računa se nova matrica -  $Q$
3. U matrici  $Q$  pronalazi se minimalna vrijednost između dva čvora
4. Pronađena dva čvora spajaju se s novim čvorom
5. Iz matrice udaljenosti brišu se dva odabrana čvora i dodaje se novi čvor
6. Ako matrica udaljenosti sadrži više od tri čvora, skoči na 2. korak
7. Završni korak spajanja zadnja tri čvora s novim čvorom

# Opis algoritma



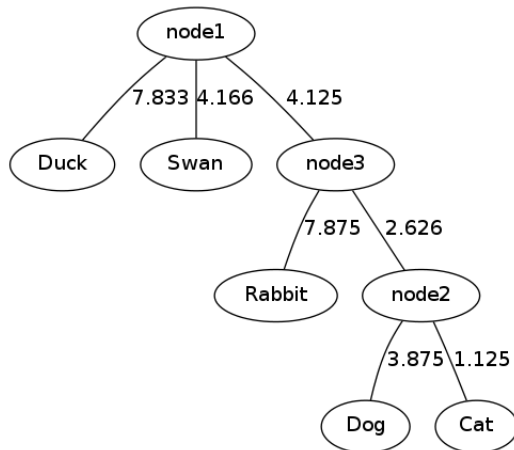
Slika: Početno stablo

# Opis algoritma



Slika: Prvi korak

# Opis algoritma



Slika: Zadnji korak

# Opis algoritma - VAŽNO!

- ▶ Računanje matrice  $Q$  i to  $N - 3$  puta kao:

$$Q(i, j) = (N - 2)d(i, j) - \sum_{k=1}^N d(i, k) - \sum_{k=1}^N d(j, k)$$

- ▶ Direktna implementacija:  $O(N^4)$
- ▶ Sume računati prije

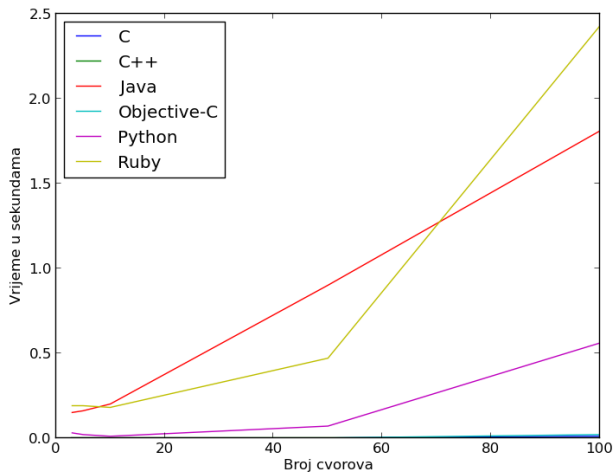
# Rezultati

- ▶ Izgrađena infrastruktura
- ▶ Računanje udaljenosti iz FASTA datoteke i povezivanje s implementacijama
- ▶ Lakše testiranje - izlaz je vizualni prikaz stabla
- ▶ Točnost - skripta koja uspoređuje dva grafa
- ▶ NJ nije deterministički!

# Rezultati

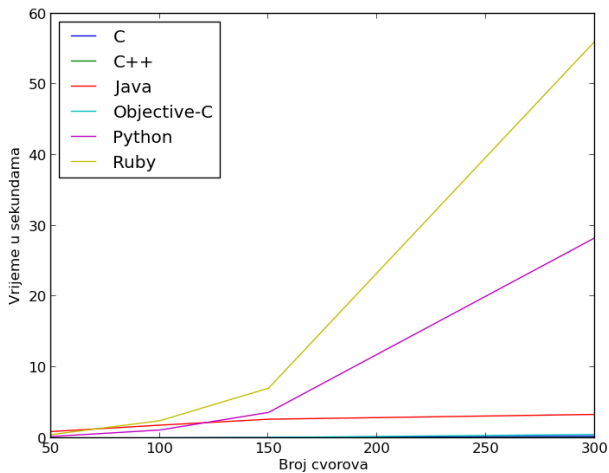
- ▶ Filip Beć - **Objective-C**
- ▶ Zorana Ćurković - **Python**
- ▶ Goran Gašić - **Java**
- ▶ Melita Kokot - **Ruby**
- ▶ Dino Šantl - **C**
- ▶ Igor Smolković - **C++**

# Vrijeme

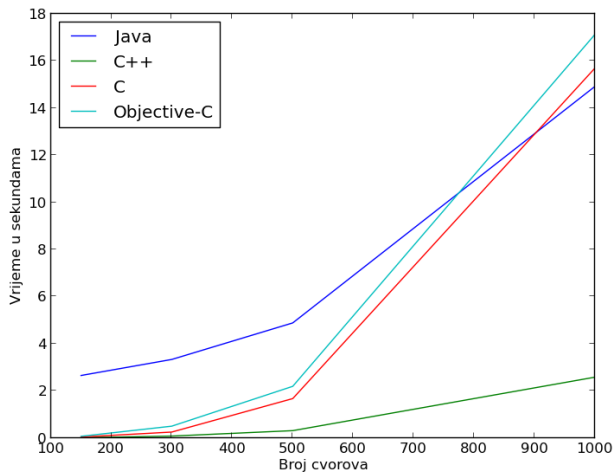




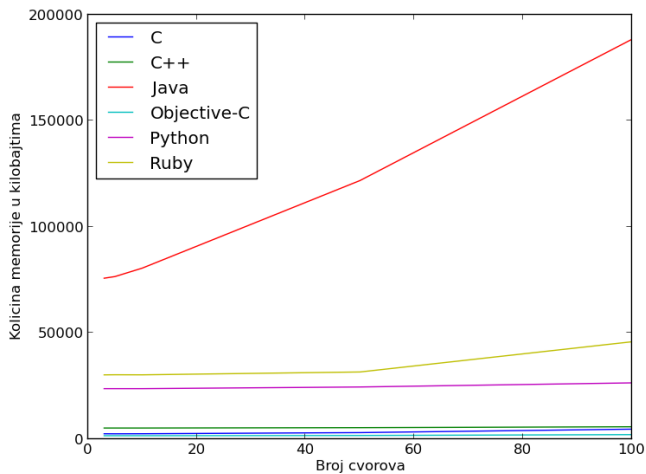
# Vrijeme



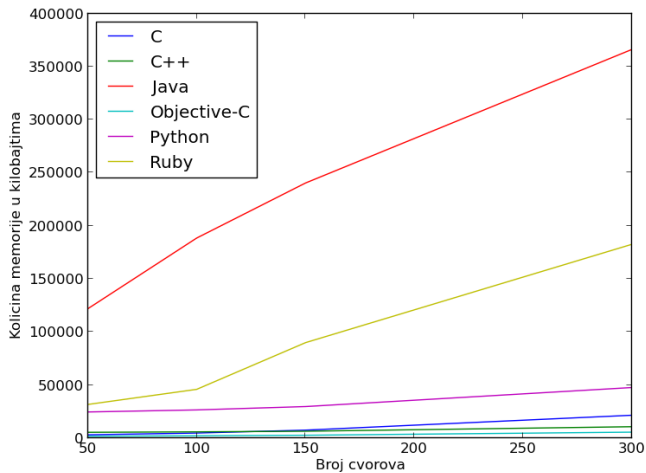
# Vrijeme



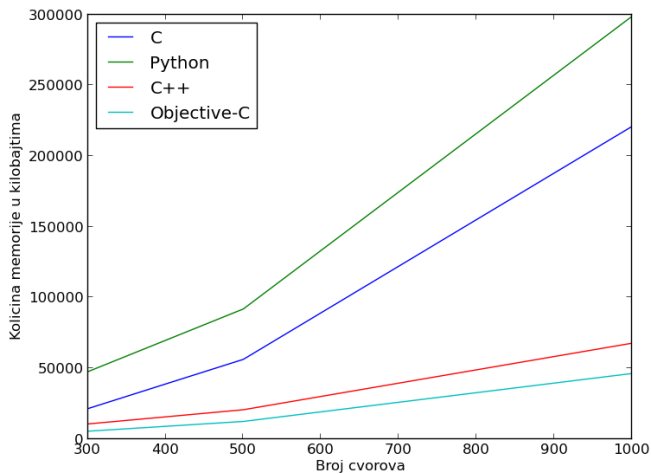
# Memorija



# Memorija



# Memorija



Hvala na pažnji!