# Projekti za 100 bodova na predmetu Bioinformatika 2018./2019.

- broj članova tima: 1-3
- implementacija: C/C++
- opis algoritma, implementacije i testiranje
- dozvoljeno je korištenje pomoćnih knjižnica u zadatcima gdje je tako navedeno, a za ostale situacije možete se dogovoriti s nastavnikom koji je zadao temu
- za svaki dan zakašnjenja umanjuje se konačan broj bodova za 3 boda

## (1) Pronalazak mutacija pomoću treće generacije sekvenciranja (RV – robert.vaser@fer.hr)

Ulaz: referentni genom i skup očitanja dobiven sekvenciranjem mutiranog genoma. Obje datoteke su u FASTA formatu.

Cilj: Za dani ulaz, pronaći razlike između referentnog genoma i sekvenciranog mutiranog genoma. Mutacije uključuju jednostruke substitucije, umetanja i brisanja. Očitanja je potrebno mapirati na danu referencu pomoću k-mer indeksa, poravnati ih te iz gomile poravnanja razlučiti mutacije. Zabranjeno je koristiti gotove implementacije.

Izlaz: Lista mutacija u odnosu na referencu (gdje je prvi nukleotid na poziciji 0), u CSV formatu kao što je prikazano u tablici ispod.

| ***Mutacija*** | **Linija u CSV datoteci** | | |
|---|---|---|---|
| *Substitucija* | X | Pozicija u referenci na kojoj se dogodila substitucija | Zamjenska nukleotidna baza |
| *Umetanje* | I | Pozicija u referenci prije koje se dogodilo umetanje | Umetnuta nukleotidna baza |
| *Brisanje* | D | Pozicija u referenci na kojoj se dogodilo brisanje | - |

Evaluacija: usporediti rezultate s referentnom implementacijom pomoću Jaccardovog indeksa. Za testne skupove, rezultate referentne implementacije i skriptu za evaluaciju potrebno se javiti nastavniku.

Bodovanje:

| | **Broj bodova** |
|---|---|
| Program<br>• ako program ne radi ispravno na testnim podatcima prilikom demonstracije umanjuje se konačan broj bodova za 10 bodova (prepravke napraviti u roku od 2 dana)<br>• vremensko ograničenje od 30min na 1 dretvi, u protivnom se oduzima 5 bodova<br>• memorijsko ograničenje od 16 GB RAM-a, u protivnom se oduzima 5 bodova<br>• točnost rezultata:<br>  ○ za odstupanje veće od 50% od referentne implementacije oduzima se 10 bodova<br>  ○ za za odstupanje veće od 75% od referentne implementacije oduzima se 25 bodova | 80 |
| Dokumentacija<br>• opis algoritma i vizualizacija na jednostavnom primjeru<br>• obavezno navesti popis literature te navesti izvore unutar teksta<br>• napraviti usporedbu točnosti, vremena izvođenja i utroška memorije vaše implementacije i izvorne | 15 |
| Prezentacija<br>• oduzimaju se bodovi, ako je prezentacija dulja od predviđenoga vremena | 5 |

Preporučena literatura:

1. Algoritmi preklapanja - skripta iz bioinformatike
2. Minimizers - https://academic.oup.com/bioinformatics/article/20/18/3363/202143

## (2) Poboljšanje djelomično sastavljenog genoma dugim očitanjima (KK – kresimir.krizanovic@fer.hr)

Cilj: Zadani genom već je djelomično sastavljen nekim od postojećih alata. Međutim, postupak sastavljanja nije bio sasvim uspješan te je rezultat fragmentiran - skup sastavljenih sekvenci (contig-a) za koje ne znamo kako se međusobno povezuju u cijeli genom. Potrebno je implementirati postupak *scaffolding*-a, koji će iskoristiti duga očitanja da bih povezao pojedine contige u dulje sekvence. Pri tome je potrebno implementirati algoritam opisan u radu:

- Huilong Du, Chengzhi Liang; Assembly of chromosome-scale contigs by efficiently resolving repetitive sequences with long reads, bioRxiv 345983; doi: https://doi.org/10.1101/345983.

Ulazni podaci:

- Skup već sastavljenih contig-a
- Skup očitanja
- Preklapanja između contig-a i očitanja u PAF formatu
- Međusobna preklapanja očitanja u PAF formatu

Izlazni podaci:

- Poboljšani skup sastavljenih contiga u FASTA formatu

Skupovi očitanja i već sastavljenih contiga bit će pripremljeni kao testni podaci. Dok će se preklapanja dobiti pomoći alata Minimap2 (https://github.com/lh3/minimap2), koristeći opciju:

```
./minimap2 -x ava-pb contigs.fa reads.fa > overlaps.paf
```

Za preuzimanje sintetskih i stvarnih testnih podataka potrebno se javiti na kresimir.krizanovic@fer.hr.

Evaluacija:

- Testiranje na sintetskim podacima i usporedba s referencom pomoću alata Gepard, dostupan na http://cube.univie.ac.at/gepard.
- Testiranje na stvarnim podacima, usporedba s referencom pomoću alata Gepard, te usporedba s referentnim rezultatima gledajući mjere:
    - Broj contig-a
    - Duljina najduljeg contig-a

Bodovanje:

| | **Broj bodova** |
|---|---|
| Program<br>• ako program ne radi ispravno na testnim podatcima prilikom demonstracije umanjuje se konačan broj bodova za 10 bodova (prepravke napraviti u roku od 2 dana)<br>• vremensko ograničenje od 60min na 1 dretvi, u protivnom se oduzima 5 bodova<br>• memorijsko ograničenje od 16 GB RAM-a, u protivnom se oduzima 5 bodova<br>• točnost rezultata:<br>  ○ ako program ne radi ispravno na sintetskim podacima oduzima se 40 bodova<br>  ○ za odstupanje veće od 25% od referentnih rezultata oduzima se 10 bodova<br>  ○ za za odstupanje veće od 50% od referentnih rezultata oduzima se 25 bodova | 80 |
| Dokumentacija<br>• opis algoritma i vizualizacija na jednostavnom primjeru<br>• obavezno navesti popis literature te navesti izvore unutar teksta<br>• napraviti usporedbu točnosti, vremena izvođenja i utroška memorije vaše implementacije i izvorne | 15 |
| Prezentacija<br>• oduzimaju se bodovi, ako je prezentacija dulja od predviđenoga vremena | 5 |

Preporučena literatura:


3. Skripta iz bioinformatike
4. PAF format: https://github.com/lh3/miniasm/blob/master/PAF.md
5. Scaffolding algoritam HERA:
   Huilong Du, Chengzhi Liang; Assembly of chromosome-scale contigs by efficiently resolving repetitive sequences with long reads, bioRxiv 345983; doi: https://doi.org/10.1101/345983.
6. Alat za DOT plot Gepard:
   Jan Krumsiek, Roland Arnold, Thomas Rattei; Gepard: a rapid and sensitive tool for creating dotplots on genome scale, Bioinformatics, Volume 23, Issue 8, 15 April 2007, Pages 1026–1028, https://doi.org/10.1093/bioinformatics/btm039.
7. Alat za računanje preklapanja Minimap2 https://github.com/lh3/minimap2

(3) **Space-efficient and exact de Bruijn graph representation based on a Bloom filter** (Chikhi and Rizk. 2013) (MDL)

- https://almob.biomedcentral.com/articles/10.1186/1748-7188-8-22

U izradi programa:

- dozvoljeno koristiti program/dijelove programa Jellyfish za brojanje k-mera
- dozvoljeno koristiti neku gotovu implementaciju Bloomovog filtera
- testirati za E. coli skup očitanja
- usporediti s originalnom implementacijom (http://minia.genouest.org/)

(4) **Improving Bloom Filter Performance on Sequence Data Using k-mer Bloom Filters** (Pellow et al 2016) (MDL)

- https://link.springer.com/chapter/10.1007/978-3-319-31957-5_10
- dozvoljeno koristiti neku gotovu implementaciju Bloomovog filtera
- usporediti s originalnom implementacijom: https://github.com/Kingsford-Group/kbf

(5) **Cuckoo Filter** (Fan et al 2013; Fan et al 2014) (MDL)

- Fan et al. 2013. "Cuckoo Filter: Better Than Bloom" (https://www.cs.cmu.edu/~binfan/papers/login_cuckoofilter.pdf)
- Fan et al. 2014. "Cuckoo Filter: Practically Better Than Bloom" (http://www.cs.cmu.edu/%7Ebinfan/papers/conext14_cuckoofilter.pdf)
- tražiti slučajne podnizove (k-mere uz različite k, npr. k = 10, 20, 50, 100, 200) u E. coli genomu
- usporediti s originalnom implementacijom: https://github.com/efficient/cuckoofilter

(6) **Ukkonenov algoritam za izgradnju sufiksnog stabla** (Ukkonen, 1995) (MDL)

- Ukkonen, On-line construction of suffix-trees (http://www.cs.helsinki.fi/u/ukkonen/SuffixT1.pdf)
- http://en.wikipedia.org/wiki/Suffix_tree
- http://en.wikipedia.org/wiki/Ukkonen%27s_algorithm
- https://stackoverflow.com/questions/9452701/ukkonens-suffix-tree-algorithm-in-plain-english/9513423#9513423

(7) **Određivanje LCP polja korištenjem modificiranog algoritma SA-IS** (Fischer, 2011) (MDL)

- Inducing the LCParray (Fischer, 2011) (http://arxiv.org/pdf/1101.3448.pdf)
- originalna implementacija: http://algo2.iti.kit.edu/english/1828.php
- novija implementacija: https://github.com/kurpicz/sais-lite-lcp
- usporediti s originalnom i novijom implementacijom

(8) **Određivanje poravnanja parova sljedova korištenjem HMM** (MDL)

- Hidden Markov Models and their Applications in Biological Sequence Analysis (Yoon, 2009) https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2766791/
- Pairwise alignment using HMMs: http://www.stat.purdue.edu/~junxie/topic4.pdf

## Bodovanje projekata (3) - (8)

| **Komponente projekta** | **Broj bodova** |
|---|---|
| Program - testiranje<br>• ako program ne radi ispravno na testnim podatcima umanjuje se konačan broj bodova za 10 bodova<br>• prepravke napraviti u roku 2 dana<br><br>Performanse programa (vrijeme izvođenja i utrošak memorije)<br>• ako se program uspoređuje sa studentskim rješenjem od prošle godine, implementacija mora biti unutar 10% vremena izvođenja i utroška memorije u odnosu na navedenu referencu za isti skup podataka (npr. ako referentni program koristi 1 GB memorije za neki skup podataka, onda Vaša implementacija treba koristiti najviše 1,1 GB memorije)<br>o oduzima se 10 bodova, ako je odstupanje do 20%<br>o oduzima se 15 bodova, ako je odstupanje veće od 20%<br><br>• ako se program uspoređuje s objavljenim rješenjem, implementacija mora biti unutar 70% vremena izvođenja i utroška memorije u odnosu na navedenu referencu (npr. ako referentni program koristi 1 GB memorije za neki skup podataka, onda Vaša implementacija treba koristiti najviše 1,7 GB memorije)<br>o oduzima se 10 bodova, ako je odstupanje do 100%<br>o oduzima se 15 bodova, ako je odstupanje veće od 100% | 60 |
| Testiranje na sintetskim podatcima $10^2$-$10^6$ znakova<br>• svi rezultati moraju biti u dokumentaciji – prikazani u tablici i/ili grafu | 10 |
| Testiranje na stvarnim podatcima (*Escherichia coli* ili po dogovoru ovisno o zadatku)<br>• svi rezultati moraju biti u dokumentaciji – prikazani u tablici i/ili grafu | 10 |
| Dokumentacija<br>• opis algoritma i vizualizacija na jednostavnom primjeru (5 bodova)<br>• obvezno navesti popis literature i navesti izvore unutar teksta (5 bodova)<br>• za svaki algoritam napraviti analizu točnosti, vremena izvođenja i utroška memorije za različite testne slučaje (5 bodova) | 15 |
| Prezentacija<br>• oduzimaju se bodovi, ako je prezentacija dulja od predviđenoga vremena (1 bod za svaku minutu prekoračenja) | 5 |