Bioinformatika

Ak. god. 2019./2020.

*Usporedba programa za mapiranje očitanja na genom*

Dokumentacija, Rev. 1

Zdravko Čićin-Šain

Fran Penić

Toma Jurčić

Martin Radin

Mateo Žaja

Datum predaje: *14.01.2020.*

Sadržaj

[1. Uvod 3](#_Toc29737650)

[2. SNAP 4](#_Toc29737651)

[2.1. Prednosti i nedostatci 4](#_Toc29737652)

[2.2. Princip rada 4](#_Toc29737653)

[2.3. Primjer rada 6](#_Toc29737654)

[3. Bwamen 10](#_Toc29737655)

[4. GraphMap2 11](#_Toc29737656)

[5. Bowtie2 12](#_Toc29737657)

[5.1. Prednosti i nedostatci 12](#_Toc29737658)

[6. Maq 13](#_Toc29737659)

[7. Zaključak 14](#_Toc29737660)

[Literatura 15](#_Toc29737661)

# Uvod

Razvojem programiranja i računalne moći, počelo se razmišljati gdje sve postoji potreba za računalima. Iako mnogima zvuči nespojivo, računala i biologija su ubrzo postala usko povezana područja.

Nakon početnog vremena prilagodne računala u biologiji, ubrzo je počela utrka u sekvenciranju. Prije par godina se činilo nedostižno, ali danas se već naširoko govori o ideji sekvenciranja za samo 1000$. Sekvenciranjem možemo otkriti pregršt informacija o nama kao vrsti, ali i kao pojedincu, stoga je stavljen velik naglasak na dostupnost jeftinog i pouzdanog sekvenciranja širokoj publici.

Ljudski genom je relativno kompliciran, stoga se u mnogim istraživačkim projektima koriste neki jednostavniji organizmi (npr. E. Coli). Razvojem softvera otvorenog koda (engl. *open-source*) mnogi alati za mapiranje očitanja su postali dostupni široj zajednici.

Glavna ideja ovog projekta je testiranje pet različitih alata te njihova međusobna usporedba. Svaki projekt zahtjeva specifičan alat koji će najbolje služiti ostvarenju cilja, stoga je svrha ovog projekta dati uvid u prednosti i mane takvih alata kako bi njihov izbor bio jasniji široj publici zainteresiranih osoba.

# SNAP

SNAP (Scalable Nucleotide Alignment Program) jedan je od najbržih dostupnih programa za mapiranje očitanja genoma. Razvijen je zajedničkim naporom timova s UC Berkeley AMP Lab, Microsoft i UCSF, a napisan je u programskom jeziku C++. Dostupan je za sve operacijske sustave, a njegovo korištenje je potpuno besplatno.

### 2.1. Prednosti i nedostatci

Glavna prednost SNAP-a je njegova brzina. SNAP je u prosjeku 3 do 20 puta brži od alata poput BWA-mem, Bowtie2 i Novoalign, a u isto vrijeme zadržava kvalitetu očitanja. SNAP je prilagođen za arhitekturu x86 procesora što mu omogućava da se izvodi na velikom broju današnjih računala. Podržan je širok spektar formata datoteka (BAM, FASTQ, gzipped FASTQ, FASTA, SAM), stoga je prikladan alat za velik broj istraživanja koja zahtijevaju barem jedan format od gore navedenih podataka.

Iako SNAP ima velik broj pozitivnih strana, on dolazi s par nedostataka. Glavni problem je izrazito velika memorijska potrošnja. SNAP prilikom svog rada koristi Hash tablice zvane *Index* tablice (objašnjeno kasnije u tekstu) kako bi postigao veliku brzinu rada. Prije svakog mapiranja genoma, potrebno je izgraditi takvu tablicu za ulaznu FASTA datoteku. Veličina same *Index* tablice ovisi o velikom broju parametra, ali najviše ovisi o samom broju ulaznih sekvenci baza. Za kompletni ljudski genom (oko 3 bilijuna baza) gradi se *Index* tablica veličine 40~57 GB. Računalo koje pokreće SNAP nad takvom *Index* tablicom trebalo bi imati barem 64 GB RAM-a *[1].* Osim ogromne količine RAM-a, računalo bi također trebalo imati što veći broj jezgara kako bi se izračun maksimalno ubrzao. Za izvođenje SNAP-a na složenijim podacima, potrebno je imati pristup izrazito jakom računalu zbog memorijskih zahtjeva. Za jednostavnije programe situacija nije toliko strašna, stoga se manje zahtjevni zadaci mogu izvoditi na manje naprednim računalima.

Drugi nedostatak zasniva se na formatu zapisa podataka. SNAP zbog načina zapisa podataka nikad neće moći izgraditi tablicu za genome koje sadrže više od 232 baza, a u praksi sve iznad 3 bilijuna baza izaziva kolizije u *Index* tablici, stoga SNAP nije primjenjiv za izrazito dugačke genome (npr. ameba).

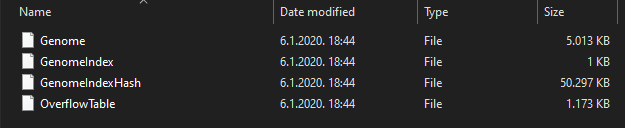
Zadnji nedostatak nije toliko nedostatak SNAP-a koliko nekonzistentnosti FASTQ formata. FASTQ format može sadržavati sekvence koje sadrže niz znakova koji se prostire u više redova. SNAP ne može učitati FASTQ datoteku u takvome formatu, stoga je potrebno takve datoteke pretprocesirati prije korištenja što oduzima dodatnu količinu vremena.

### 2.2. Princip rada

Kao što je već ranije spomenuto u tekstu, SNAP se bazira na *Index* tablicama. Prilikom pokretanja programa, potrebno je predati ulaznu FASTA datoteku uz pomoć koje će se stvoriti *Index* tablica. U komandnoj ljusci potrebno je pozvati naredbu

* .\snap index ulaz.fa index-dir

Prethodna naredba gradi *Index* tablicu u direktoriju naziva *index-dir* (stvori se ako prethodno ne postoji).



Slika 1 – Sadržaj index-dir direktorija

Slika 1 prikazuje sadržaj *index-dir* direktorija. Glavna datoteka koju SNAP koristi za prilikom svog rada je *GenomeIndexHash*. Veličina ulazne datoteke je ~5 MB, a veličina te tablice je 10 puta veća (~50 MB). Za male ulazne podatke nema problema prilikom pokretanja programa, problem dolazi ako želimo raditi kompleksnije izračune jer veličina *Index* tablice drastično raste.

Postavlja se pitanje zašto su nam uopće potrebne *Index* tablice i na koji način one služe programu prilikom izračuna.

Glavna prednost hash tablice je upravo brzina kojom možemo pristupiti svakom pojedinom podatku, a upravo to je glavna ideja za njihovo korištenje. Početno se izgradi velik set podataka, ali taj set podataka koristimo za mapiranje očitanja i značajno skraćujemo vrijeme izvođenja programa. Npr. ako imamo ulaznu sekvencu nekog kromosoma X

0123456789012345678901234

AAGCTTCTCACCCTGTTCCTGCATA

i želimo izgraditi *Index* s parametrom *seed-size* = 20, onda će *Index* tablica imati podatke

0123456789012345678901234

AAGCTTCTCACCCTGTTCCT - kromosom X, lokacija 0

AGCTTCTCACCCTGTTCCTG - kromosom X, lokacija 1

GCTTCTCACCCTGTTCCTGC - kromosom X, lokacija 2

CTTCTCACCCTGTTCCTGCA - kromosom X, lokacija 3

TTCTCACCCTGTTCCTGCAT - kromosom X, lokacija 4

TCTCACCCTGTTCCTGCATA - kromosom X, lokacija 5

Ako iz FASTQ datoteke pročitamo slijed CTTCTCACCCTGTTCCTGCA, SNAP će odmah prepoznati da se radi o kromosomu X, lokacija 3.

Ako iz FASTQ pročitamo slijed CTTCTATCCCTGTTCCTGCA, SNAP neće pronaći odgovarajući zapis u *Index* tablici, odnosno javit će da nije pronašao odgovarajuću sekvencu. SNAP očekuje da slijed koji pročita savršeno odgovara nekome od zapisa iz tablice. Ako povećamo parametar *seed-size*, onda će postojati manja vjerojatnost da će ulazni slijed biti identičan nekom od zapisa iz *Index* tablice. Manji *seed-size* uzrokuje da SNAP pronađe više zapisa iz *Index* tablice kojima ulazni slijed odgovara. Takva situacija za posljedicu ima povećanje vremena izvođenja jer SNAP mora prvo odlučiti koji će zapis iz tablice odabrati, stoga je taj parametar najbolje ostaviti napredefiniranoj vrijednosti 20 osim ako korisnik zbilja zna kako treba podesiti taj parametar da dobije najbolje rezultate. SNAP podržava *seed-size* do veličine 32. Za veličinu 23 i ~100 pročitanih baza u ulaznoj sekvenci, pogreška iznosi oko 2%.

Nakon što smo izgradili *Index* tablicu, napokon možemo krenuti s mapiranjem očitanja.

SNAP nudi dvije mogućnosti pokretanja programa:

* Pokretanje s jednom ulaznom datotekom (engl. *single unpaired read*)
* Pokretanje s dvije ulazne datoteke (engl. *paired end read*)

Ako program pokrećemo s jednom ulaznom datotekom u FASTQ formatu, potrebno je pozvati naredbu iz komandne ljuske

* .\ snap single index-dir ulaz.fq -o output.sam

Izlazna datoteka može biti u *.sam* ili *.bam* formatu. SNAP će generirati izlaznu datoteku i u komandnoj ljusci ispisati izvještaj.

Slično je pokretanje s dvije ulazne datoteke. Potrebno je pozvati naredbu iz komandne ljuske

* .\ snap paired index-dir ulaz1.fq ulaz2.fq -o output.sam

Postoji velik broj parametara koji se može koristiti za poboljšanje performansi, ali to je područje preopsežno, stoga se čitatelja upućuje na čitanje službene dokumentacije iz izvora *[1]* ako ga to zanima.

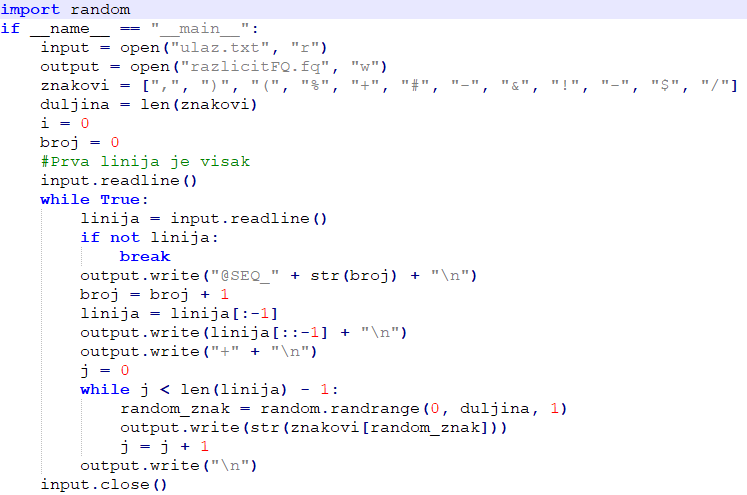
### 2.3. Primjer rada

Za demonstraciju rada programa, koristit ćemo četiri ulazna slučaja:

* Kratak dio sekvence genoma E. Coli (~2500 baza, FASTQ format)
* Kompletna sekvenca genoma E. Coli (~5 196 800 baza, FASTQ format)
* Slučajno generirana sekvenca genoma (~5 196 800 baza, FASTQ format)
* Kompletna sekvenca genoma E. Coli i slučajno generirana sekvenca genoma (~5 196 800 baza, FASTQ format

Određene FASTQ datoteke nisu u formatu koju SNAP prihvaća, stoga je potrebno pretprocesirati ulaznu datoteku.

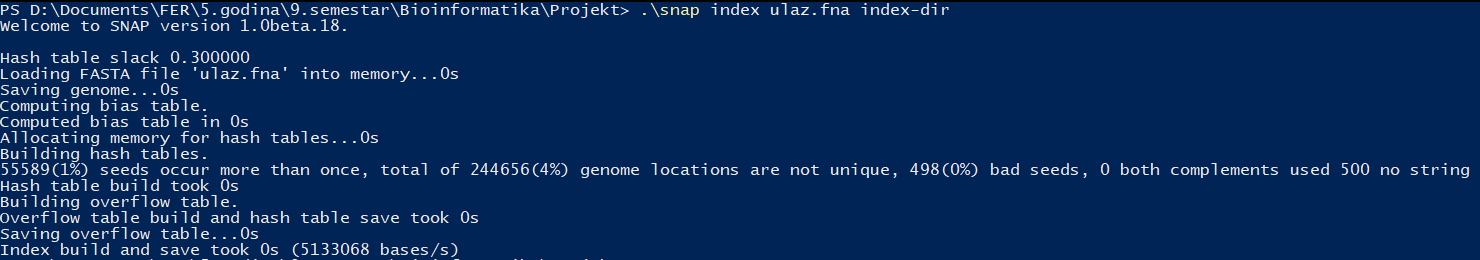
Slučajno generirane sekvence napravljene su pomoć Python skripte.



Slika 2 – Python skripta za generiranje sekvence genoma

Skripta uzima ulaznu datoteku koja sadrži sekvencu genoma, svakoj sekvenci pridaje vrijednost, odnosno kvalitetu te stavlja pripadajuće oznake. Takvim postupkom generiramo izlaznu datoteku u FASTQ formatu.

*Index* tablica se koristi prilikom svakog mapiranja, stoga ju je potrebno napraviti na početku.



Slika 3 – Rezultat izgradnje Index tablice

*Index* tablica zbog relativno kratkog genoma E. Coli izgradi se za manje od jedne sekunde, a zauzima ~56 MB prostora.

1. Kratak dio sekvence genoma E. Coli

Nakon generiranja *Index* tablice pokrećemo SNAP i izvršavamo naredbu

* .\ snap single index-dir ulazSlicni.fq -o output.sam



Slika 4 – Rezultat pokretanja programa (prvi slučaj)

Slika 4 prikazuje rezultat pokretanja SNAP-a na kratkoj sekvenci genome E. Coli. Rezultati izvođenja prikazani su u tablici 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ukupan broj čitanja** | **Poravnato** | **Različito** | **Memorija (MB)** | **Vrijeme izvođenja (s)** |
| 8 | 100% | 0% | 56 MB | < 1 s |

Tablica 1 – Tablični rezultati pokretanja programa (prvi slučaj)

Rezultati su očekivani. Kratka sekvenca genome E. Coli savršeno je poravnata s ulaznim genomom. Vrijeme izvođenja je kraće od jedne sekunde, a memorije je ~56 MB jer se zbog veće brzine izvođenja *Index* tablica direktno učita u RAM (ne radi se čitanje s diska).

Kao izlaz generirana je *output.sam* datoteka koja može koristiti za detaljnije analize.

1. Kompletna sekvenca genoma E. Coli

Izvršavamo naredbu

* .\ snap single index-dir slicniFQ.fq -o output.sam



Slika 5 – Rezultat pokretanja programa (drugi slučaj)

Slika 5 prikazuje rezultat pokretanja SNAP-a na kompletnoj sekvenci genome E. Coli. Rezultati izvođenja prikazani su u tablici 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ukupan broj čitanja** | **Poravnato** | **Različito** | **Memorija (MB)** | **Vrijeme izvođenja (s)** |
| 64 151 | 94.98% | 0% | 67 MB | < 1 s |

Tablica 2 – Tablični rezultati pokretanja programa (drugi slučaj)

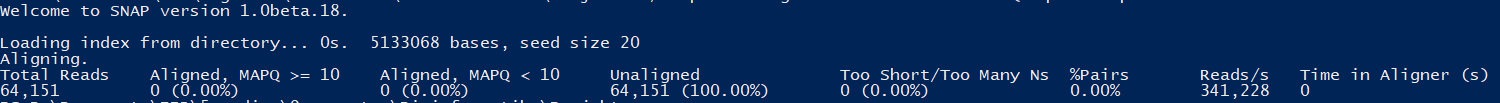
MAPQ nam govori o kvaliteti pročitane sekvence. Zapisi koji imaju MAPQ > 10 se mogu smatrati pouzdano očitanima, a oni koji imaju MAPQ < 10 nisu u potpunosti pouzdani. Slika 5 nam govori da je 94.98% zapisa poravnato, ali 5.02% zapisa nije pouzdano poravnato. Takva situacija se događa kada FASTQ datoteka sadrži baze koje imaju lošu ocjenu očitanja, stoga SNAP ne može u potpunosti zaključiti što se nalazi na tome mjestu. SNAP će pokušati naći najbolju opciju i ponašati se kao da je to očitano (opcija koja prolazi najbolje poravnanje).

Vrijeme izvođenja je ostalo gotovo identično u usporedbi s prvim slučajem, ali su se memorijski zahtjevi malo povećali zbog veličine ulazne datoteke.

1. Slučajno generirana sekvenca genoma

Python skriptom prikazanom na slici 2 generiramo FASTQ datoteku koju pokrećemo pomoću SNAP-a naredbom

* .\ snap single index-dir razlicitFQ.fq -o output.sam



Slika 6 – Rezultat pokretanja programa (treći slučaj)

Slika 6 prikazuje rezultat pokretanja SNAP-a na slučajno generiranoj sekvenci genoma. Rezultati izvođenja prikazani su u tablici 3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ukupan broj čitanja** | **Poravnato** | **Različito** | **Memorija (MB)** | **Vrijeme izvođenja (s)** |
| 64 151 | 0% | 100% | 67 MB | < 1 s |

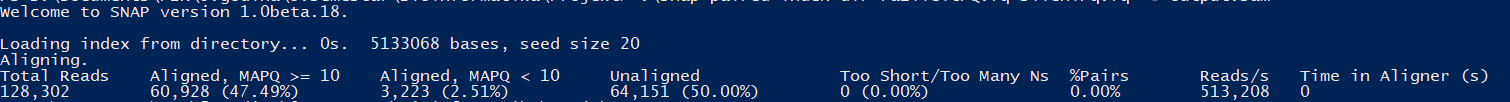
Tablica 3 – Tablični rezultati pokretanja programa (treći slučaj)

Kao što je i bilo očekivano, slučajno generirana sekvenca nema poklapanja s originalnom sekvencom, stoga vidimo da SNAP nije pronašao poravnanja. Ostali rezultati su identični kao iz prethodnog pokretanja programa.

1. Kompletna sekvenca genoma E. Coli i slučajno generirana sekvenca genoma

Izvršavamo naredbu

* .\snap paired index-dir razlicitFQ.fq slicniFq.fq -o output.sam



Slika 7 – Rezultat pokretanja programa (četvrti slučaj)

Slika 7 prikazuje rezultat pokretanja SNAP-a na dvije sekvence genoma. Rezultati izvođenja prikazani su u tablici 4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ukupan broj čitanja** | **Poravnato** | **Različito** | **Memorija (MB)** | **Vrijeme izvođenja (s)** |
| 128 302 | 47.49% | 50% | 78 MB | < 1 s |

Tablica 4 – Tablični rezultati pokretanja programa (četvrti slučaj)

Vidimo da se i u ovome slučaju vrijeme izvođenja nije promijenilo. Korištenje memorije blago je poraslo, ali ništa drastično.

Glavni nedostatak SNAP-a u smislu statističkih podataka je to što ne pokazuje sve bitne podatke. Vrijeme bi moglo biti mjereno u *ms* radi veće preciznosti i bilo bi poželjno imati ispis greške u postocima. Taj podatak, prema izvoru *[1],* ne bi trebao biti veći od 2% ako su svi parametri ispravno složeni.

Na prethodna četiri primjera demonstrirali smo glavnu prednost SNAP-a, a to je upravo njegova brzina. Ako se žele raditi kompleksnija mjerenja, onda je potrebno imati jače računalo, ali za uporabu na manje kompleksnim projektima SNAP je izvrstan izbor jer nudi izrazito brza i točna očitanja.

# Bwamen

# GraphMap2

# Bowtie2

Bowtie2 je brzi i memorisjki učikoviti alat za poravnavanje očitanja genoma. Razvijen od strane autora: Langmead B, Wilks C, Antonescu V, Charles R., Salzberg SL., čiji radovi su dostupni na linkovima priloženim u literaturi *[6]* i *[7].* Bowtie2 se distribuira pod GPLv2 *[10]* licencom i koristi se preko komandne linije kod Windows, Mac OS X te Linux operativnog sustava.

### 5.1. Prednosti i nedostatci

Bowtie2 je posebno je dobar u poravnavanju očitavanja od oko 50 do stotinjak znakova pa sve do relativno dugačkih genoma (np. genoma sisavaca). Program koristi indeksiranje pomoću FM Indexa (slično sufiksnim poljima) *[8]* (*koji je baziran na Burrows-Wheeler Transformatu ili skraceno BWT [9]* ) kako bi osigurao malo zauzeče memorije, na primjer za ljudski genom Bowtie2 koristi oko 3.2 gigabyta radne memorije računala. Program podržava modove poravnavanja: razmaknuto poravnavanje , lokalno poravnavanje i poravnavanje uparenih krajeva. Također podržava korištenje više procesora (te time i više jezgreno) za veče brzine poravnavanja.

Bowtie2 namijenjen je za usklađivanje relativno kratkih sekvenici očitanja pa sve do dugačkih genoma. Može se koristiti za proizvoljno male referentne nizove (npr. Amplikone, eng. Amplicons *[11])*, a i vrlo velika očitanja (npr. u veličinama desetaka do stotinjaka kilobaza) iako je spor kod takvih primjena. Optimiziran je za duljine očitanja i pogreške dobivenih od tipičnih illumina sekvencera *[12]*. Bowtie2 ne podržava poravnavnaje očitanja u prostoru boja *(eng. colorspace reads).*

# Maq

# Zaključak

Napisati nešto...

# Literatura

Snap

[1] <http://snap.cs.berkeley.edu/downloads/snap-1.0beta-manual.pdf>

[2] <https://www.ebi.ac.uk/ena/data/view/SRX5309760>

[3] <https://www.ncbi.nlm.nih.gov/genome/167?genome_assembly_id=161521>

[4] <http://snap.cs.berkeley.edu/downloads/snap-1.0beta-quickstart.pdf>

[5] <https://github.com/PacificBiosciences/DevNet/wiki/E-coli-K12-MG1655-Resequencing>

Bowtie2

[6] <https://academic.oup.com/bioinformatics/article/35/3/421/5055585>

[7] <https://www.nature.com/articles/nmeth.1923>

[8] <https://en.wikipedia.org/wiki/FM-index>

[9] <https://en.wikipedia.org/wiki/Burrows%E2%80%93Wheeler_transform>

[10] <http://www.gnu.org/licenses/gpl-3.0.html>

[11] <https://en.wikipedia.org/wiki/Amplicon>

[12] <https://www.illumina.com/systems/sequencing-platforms.html>

[13]

[14]

[15]