

Niniejsze opracowanie zostało stworzone przez dr Magdę Mielczarek, pracownika Uniwersytetu Przyrodniczego we Wrocławiu w ramach wykonywania obowiązków związanych z kształceniem studentów i jest przeznaczone dla studentów Biologii (Wydział Biologii i Hodowli Zwierząt) na potrzeby dydaktyczne bez prawa do dalszego rozpowszechniania.

ELEMENTY PROGRAMOWANIA

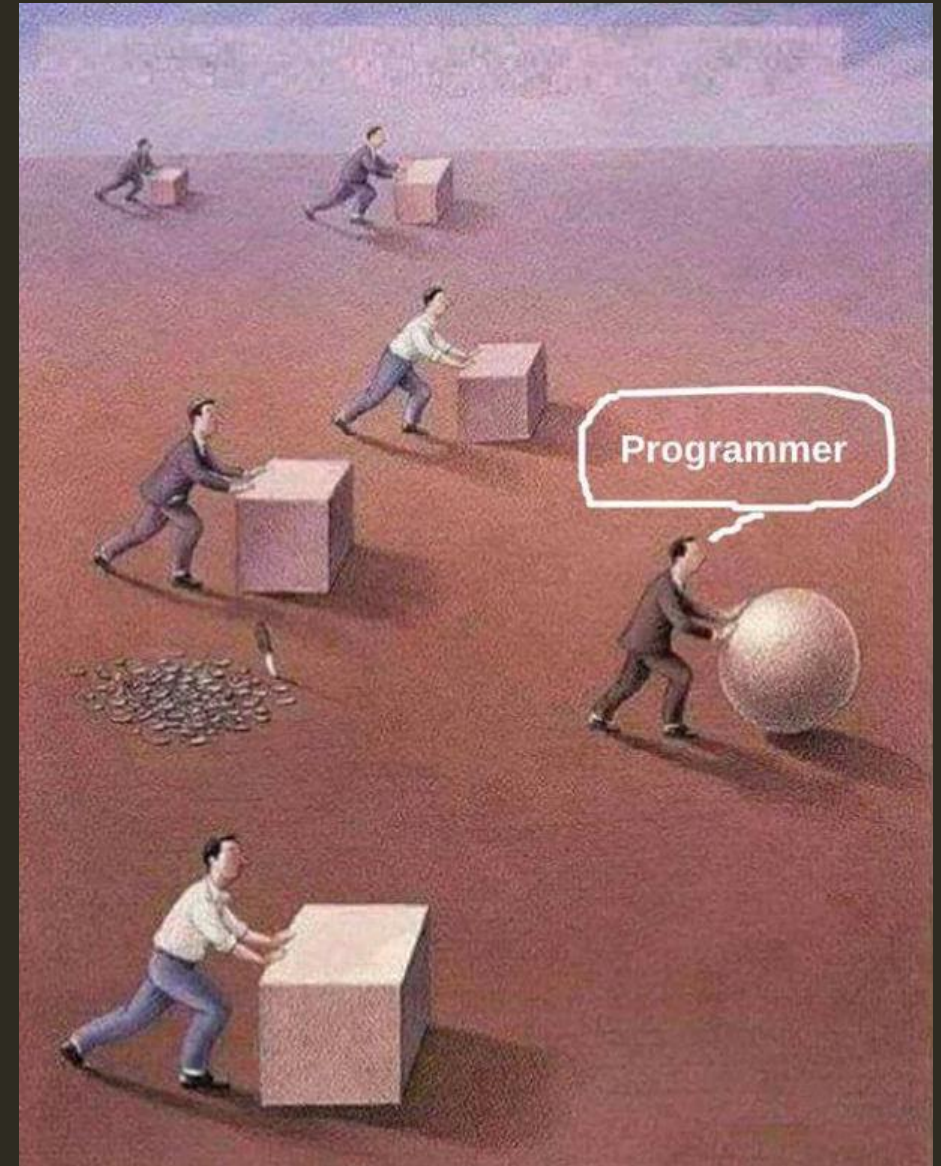
Podstawy Bioinformatyki
wykład 7

PROGRAMOWANIE W BIOINFORMATYCE

Programowanie:

- umiejętność wydawania poleceń w zrozumiałym dla komputera języku
- ułatwia pracę, automatyzuje ją i pozwala spojrzeć na (ogromnych rozmiarów) dane z innej perspektywy

**DON'T WORK HARD.
WORK INTELLIGENT.**



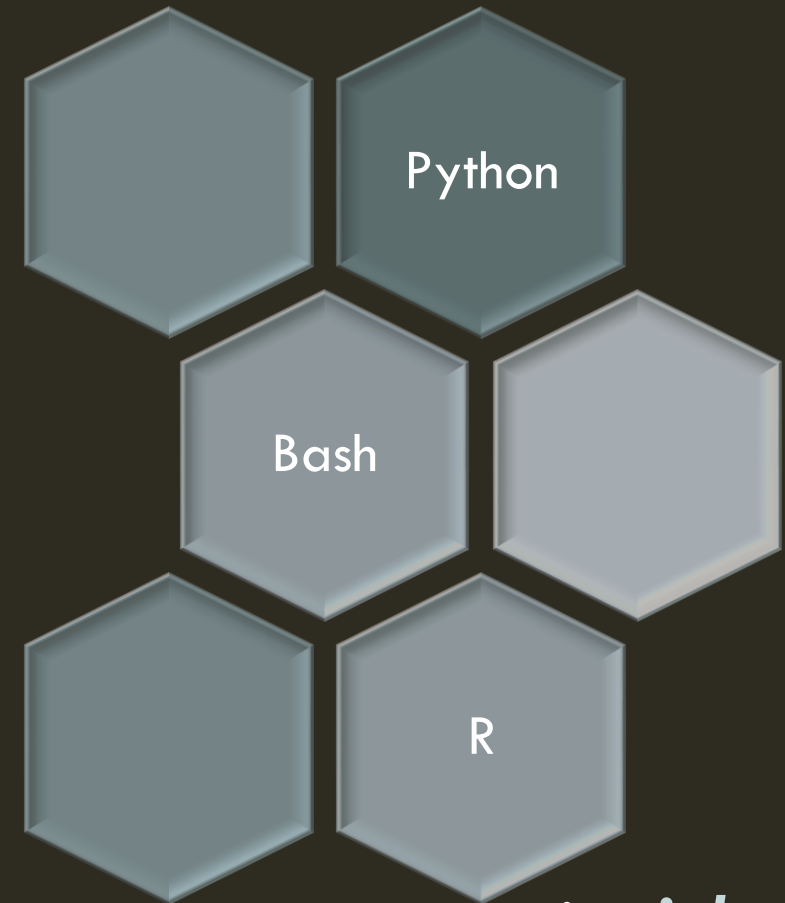
PROGRAMOWANIE W BIOINFORMATYCE

Programowanie:

- umiejętność wydawania poleceń w zrozumiałym dla komputera języku
- ułatwia pracę, automatyzuje ją i pozwala spojrzeć na (ogromnych rozmiarów) dane z innej perspektywy

W bioinformatyce przydatne są m.in:

- **Python** - język programowania
- **Bash** - środowisko programowania, język programowania
- **R** – pakiet do analizy statystycznej, język programowania
- i **wiele** innych...

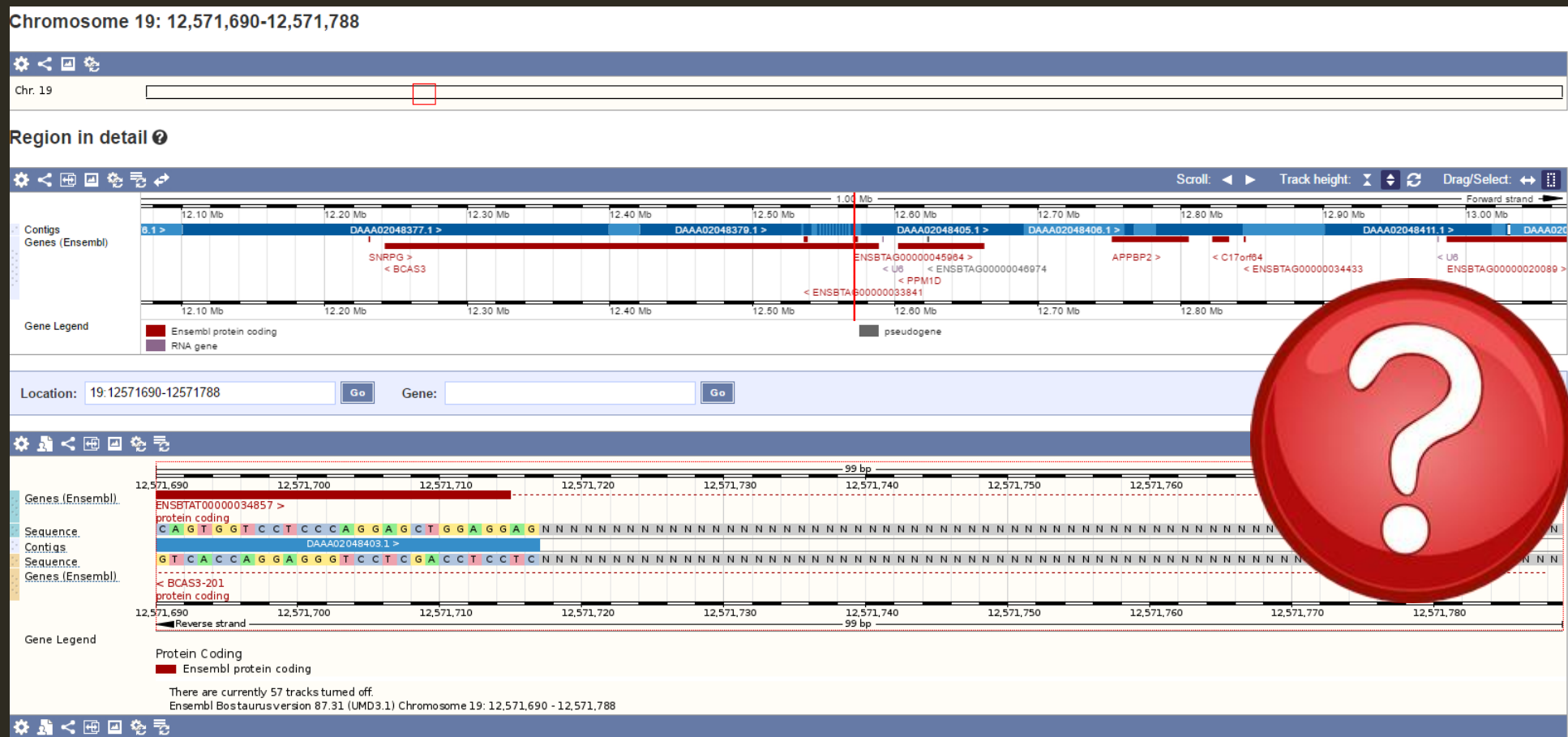


i **wiele**
innych...

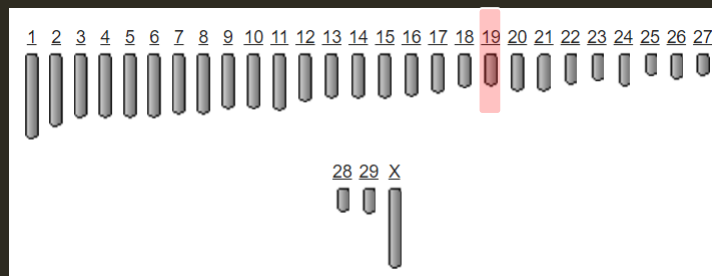
PYTHON



PROBLEM 1: JAKA JEST LICZBA N W GENOMIE?



ANY BASES



Indeks – /pub/release-87/fasta/bos_taurus/dna/

Nazwa	Rozmiar	Data modyfikacji
[katalog główny]		
Bos_taurus.UMD3.1.dna.chromosome.1.fa.gz	45.6 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.10.fa.gz	30.1 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.11.fa.gz	31.0 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.12.fa.gz	26.2 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.13.fa.gz	24.4 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.14.fa.gz	24.3 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.15.fa.gz	24.5 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.16.fa.gz	23.6 MB	08.12.2016, 15:29:00
Bos_taurus.UMD3.1.dna.chromosome.17.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.18.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.19.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.20.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.21.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.22.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.23.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.24.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.25.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.26.fa.gz		
Bos_taurus.UMD3.1.dna.chromosome.27.fa.gz		

Bos taurus breed Hereford chromosome 19, Bos_taurus_UMD_3.1.1, whole genome shotgun sequence

NCBI Reference Sequence: AC_000176.1

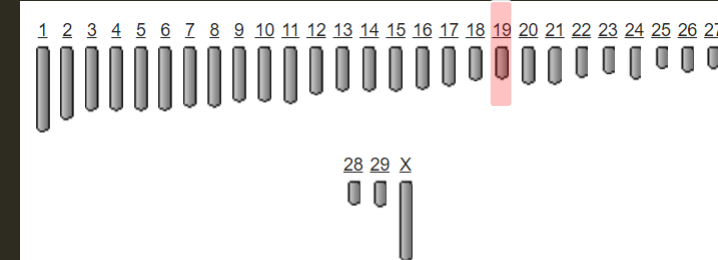
[FASTA](#) [Graphics](#)

Go to: ☐

LOCUS AC_000176 64057457 bp DNA linear CON 26-JAN-2016
DEFINITION Bos taurus breed Hereford chromosome 19, Bos_taurus_UMD_3.1.1, whole genome shotgun sequence.
ACCESSION AC_000176 GPC_000000188
VERSION AC_000176.1
DBLINK BioProject: [PRJNA33843](#)
BioSample: [SAMN02898106](#)
Assembly: [GCF_000003055.6](#)
KEYWORDS WGS; RefSeq.
SOURCE Bos taurus (cattle)
ORGANISM Bos taurus

```
>19 dna:chromosome chromosome:UMD3.1:19:1:64057457:1 REF
ACGTGCACTCTCCTGAGACGCTGAGCGGAAAGGGTTCATCTTGCGATGACGGGAAACAAC
GTGGTTTTTCTCGAGTTGCGGCGGGATTCTCGAGTTACGACGGGGAATTCAGGATGCCTC
TTGTGTTGGCCAGGGAAGTCAATCTTCCATTGAGTTGCGAAGGAAAGCTGGGGATTG
CTCTCGAGTGACTGCAGGGCCAATAGACCTCATCTAGGCTTGTGTCCAGAAGCCGAGCTT
CCTCTCCAGGGGCGACCGGGATCTCGGGGTTGCATTCCAGACGCACCCGGGGAGACAGGC
ATTCATCTCGAGTGGAAGCAAAGAACCCCGCTCTGCTCTCGAGTCGCGACGGGTATCTCT
TGGAGCTCACTGGGTGGACTAAAGGGAGTCAAGCCTCCTGAGGCGTTTGGAGAGAGGTCG
CGAGATTGGCCTCTAGGCCATGAAGGAGACGAAGGTCCTCATCTCCGCGATGACGGGGGA
GTCTCGTGGTGGTCTCGAGCGGTGGCCCCAGGGTGGGTTTCTCACGAGGTACGACGGC
GAGGTCAGTGAGCCTCTCGTGGGGCGCCAGGGAAGTCGGGTCTCCATGCGAGTGGCGAGG
GGGAGCACGTCAATTGCTCTCGAGTCATGGTAGGGGAATCTGGCCTCGAGACGTGTTGAAG
AGCGCTCTCGACCGCTTCTCGCGCTTCAGGCAAGGAAACCCTGGCTTCCCTCGACTTGTG
TGAGAAAGTCAGGGAACTGGAGGTGGG
ATTGGAAGGGGCTCATCTCCAGTTG
ACTCCGAACCGCGGGTCCCTGCAGACTG
AGGCATGGAACCTCCGCTTGCCTCTCGAG
GTATTTGGAACCTAGGGGTTTTTTCGG
TTGACTTCATTACAGGGTGGAGTTTCG
AGGGACCGGGAAATCGGGGTGCTTCGC
ATGTCGTCTGAGACCGGCCTCAACCCG
AGAAAGCAGGGGAGTCGACCCTCCTGT
AGTGGTGCCGGAAAACCTCGGTGTTCTT
GGGGGCGATCAAGGGTGCCAACTACCG
TGAGACGCTGTAGCGGGAAGGGCTTCA
TCGAGTTGCGGCGGGATTCTCGAGTTA
NNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNGCATCAAGGTGTCAAGTA
```


ANY BASES

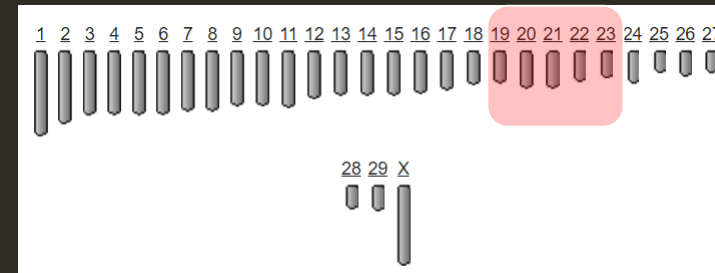


```
mielczarek@ui.inula.man.poznan.pl: ~  
### Podstawy Bioinformatyki ###  
### dla kierunku Biologia ###  
  
chrom=open('/home/mielczarek/genomes/bovine/Bos_taurus.UMD3.1.dna.chromosome.19.fa' , 'r')  
chrom.readline()  
n=0  
o=0  
  
for line in chrom:  
    for base in line:  
        if base == 'N':  
            n+=1  
        elif base == 'A' or base == 'C' or base == 'T' or base == 'G':  
            o+=1  
  
procent=(round(n)/round(n+o))*100  
  
print('Dlugosc BTA19: ' + str(n+o) + ' pz')  
print('Liczba N: ' + str(n) + ' pz')  
print('Procent N: ' + str(procent))
```

```
Dlugosc BTA19: 64057457 pz  
Liczba N: 500254 pz  
Procent N: 0.780945768734
```

```
>19 dna:chromosome chromosome:UMD3.1:19:1:64057457:1 REF  
ACGTGCACTCTCCTGAGACGCTGAGCGGAAAGGGTTCATCTTGCGATGACGGGAAACAAC  
GTGGTTTTTCTCGAGTTGCGGCGGGATTCTCGAGTTACGACGGGGAATTCAGGATGCCTC  
TTGTGTTGGCCCAGGGAAGTCGAATCTTCCATTTCGAGTTGCGAAGGAAAGCTGGGGATTG  
CTCTCGAGTGAAGTGCAGGGCCAATAGACCTCATCTAGGCTTGTGTCCAGAAGCCGAGCTT  
CCTCTCCAGGGGCGACCGGGATCTCGGGGTTGCATTCCAGACGCACCCGGGGAGACAGGC  
ATTCATCTCGAGTGGAAGCAAAGAACCCCGCTCTGCTCTCGAGTCGCGACGGGTATCTCT  
TGGAGCTCACTGGGTGGACTAAAGGGAGTCAAGCCTCCTGAGGCGTTTGGAGAGAGGTTCG  
CGAGATTGGCCTCTAGGCCATGAAGGAGACGAAGGTCCTCATCTCCGCGATGACGGGGGA  
GTCTCGTGGTGGTTCTCGAGCGGTGGCCCCAGGGTGCGGTTTCTCACGAGGTACGACGGC  
GAGGTCAGTGAGCCTCTCGTGGGGCGCCAGGGAAGTCGGGTCTCCATGCGAGTGGCGAGG  
GGGAGCACGTCAATTGCTCTCGAGTCATGGTAGGGGAATCTGGCCTCGAGACGTGTTGAAG  
AAGGCCTCTCGAGGCCTTTCTCGGGTTGAGGCAGGAAACCCTGGCTTCCCTCGACTTGTG  
CAGGTGACCTCAGGGGACTTCTCATGGTGGCTCTGAGAAGTCAGGGAAGTGGAGGTGGG  
AGGGGCCTCTCGGGACTCCACTGGGTTTGGTGCATTGGAAGGGGCCTCATCTCCAGTTG  
AGGCAGGAACCTCAGGGTTCCTCTGATTTCACTCCGAACGCCGGGTCCCTGCAGACTG
```


ANY BASES



mielczarek@ui.inula.man.poznan.pl: ~

```
### Podstawy Bioinformatyki ###
### dla kierunku Biologia ###

for x in range(19, 24):
    print('CHROM ' + str(x))
    chrom=open('/home/mielczarek/genomes/bovine/Bos_taurus.UMD3.1.dna.chromos
    chrom.readline()
    n=0
    o=0
    for line in chrom:
        for base in line:
            if base == 'N':
                n+=1
            elif base == 'A' or base == 'C' or base == 'T' or base == 'G':
                o+=1
        procent=(round(n)/round(n+o))*100

    print('Dlugosc:          '+ str(n+o) + ' pz')
    print('Liczba N:          ' + str(n) + ' pz')
    print('Procent N:           ' + str(procent))
```

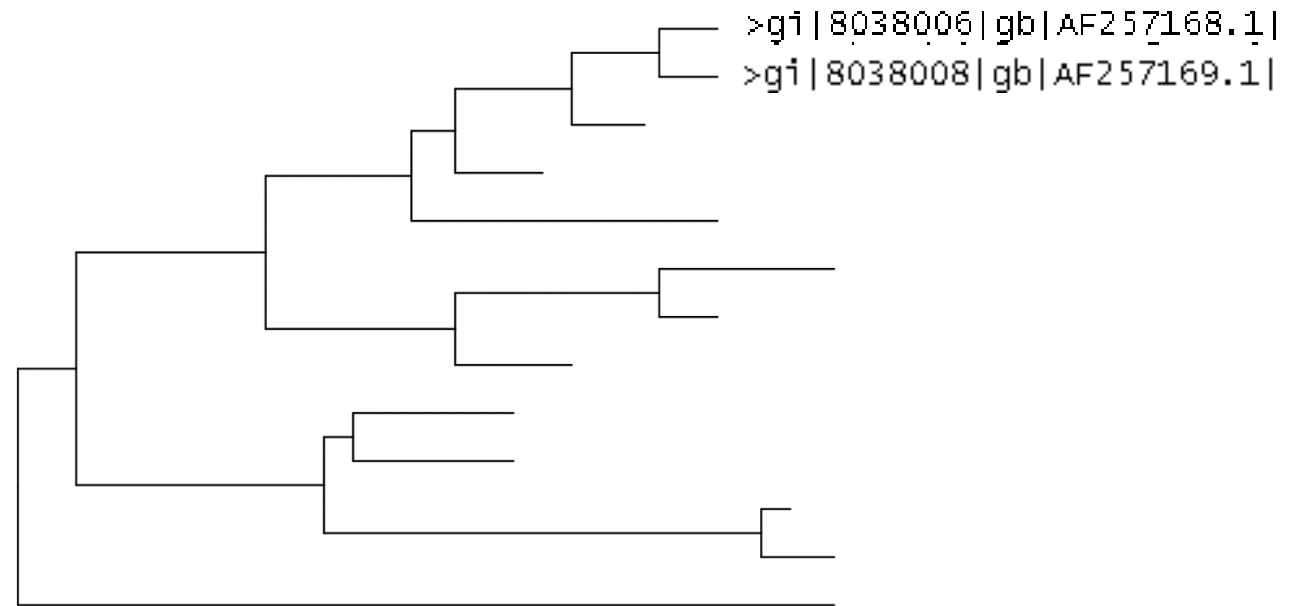
```
CHROM 19
Dlugosc:          64057457 pz
Liczba N:          500254 pz
Procent N:         0.780945768734
CHROM 20
Dlugosc:          72042655 pz
Liczba N:          491142 pz
Procent N:         0.681737784372
CHROM 21
Dlugosc:          71599096 pz
Liczba N:          546211 pz
Procent N:         0.762874156959
CHROM 22
Dlugosc:          61435874 pz
Liczba N:          358308 pz
Procent N:         0.583222760044
CHROM 23
Dlugosc:          52530062 pz
Liczba N:          395218 pz
Procent N:         0.752365378895
```

PROBLEM 2: JAK ZMIENIĆ ID SEKWENCJI?



```
>gi|8038006|gb|AF257168.1| Ursus maritimus isolate CH542 cytochrome b gene, partial cds;  
mitochondrial gene for mitochondrial protein  
GAGGGGGCTTTTCCGTAGATAAGGCGACTCTAACACGAT  
CATCCTAGCACTAGCAGCAGTCCACCTATTGTTCTCTACA  
CCATCTGACTCAGACAAAATCCCATTCATCCATACTAT  
TCACCCTAGCCTTAGCAACCCTAGTCCTATTCTCGCCCC  
CGCAAAT
```

```
>gi|8038008|gb|AF257169.1| Ursus maritimus isolate CH542 cytochrome b gene, partial cds;  
mitochondrial gene for mitochondrial protein  
GAGGGGGCTTTTCCGTAGATAAGGCGACTCTAACACGAT  
CATCCTAGCACTAGCAGCAGTCCACCTATTGTTCTCTACA  
CCATCTGACTCAGACAAAATCCCATTCATCCATACTAT  
TCACCCTAGCCTTAGCAACCCTAGTCCTATTCTCGCCCC  
CGCAAAT
```



ZMIANA ID SEKWENCJI

```
>gi|8038006|gb|AF257168.1| Ursus maritimus isolate CHS42 cytochrome b gene, partial cds; mitochondrial gene for mitochondrial product
```

```
GAGGGGGCTTTTCCGT  
CATCCTAGCACTAGCA  
CCATCTGACTCAGACA  
TCACCCTAGCCTTAGC  
CGCAAAT
```

mielczarek@ui.inula.man.poznan.pl: ~

```
plik = open('Sekwencje.fasta')  
try:  
    zmienna = plik.read()  
finally:  
    plik.close()
```

```
>gi|8038008|gb|AF257168.1| Ursus maritimus isolate CHS42 cytochrome b gene, partial cds; mitochondrial gene for mitochondrial product  
GAGGGGGCTTTTCCGT  
CATCCTAGCACTAGCA  
CCATCTGACTCAGACA  
TCACCCTAGCCTTAGC  
CGCAAAT
```

```
x=raw_input('Podaj nazwe pliku, do ktorego bedzie zapisany wynik programu: ').+'.fasta'  
plik2=open(x, 'w')
```

```
s="\n"  
o=" "
```

```
zmienna = zmienna.split('\n')  
for i in range(len(zmienna)-1):  
    if zmienna[i] != '':
```

```
>Ursus_maritimus_CHS42
```

```
>gi|8038006|gb|AF257168.1| Ursus maritimus isolate CHS42 cytochrome b gene, partial cds; mitochondrial gene for mitochondrial product
```

```
GAGGGGGCTTTTCCGTAGATAAGGCGACTCTAACACGATTCTTTGCTTTCCACTTTATTCTCCCGTTCAT  
CATCCTAGCACTAGCAGCAGTCCACCTATTGTTCCCTACACGAAACAGGATCCAACAACCCCTCTGGAATC  
CCATCTGACTCAGACAAAATCCCATTCCATCCATACTATACAATTAAGGATATTCTAGGCGCCCTACTTC  
TCACCCTAGCCTTAGCAACCCTAGTCCTATTCTCGCCGACTTACTAGGAGACCCTGATAACTATATCCC  
CGCAAAT
```

Nowy
identyfikator
sekwencji

```
+ o.join(zmienna[i]))
```



The fundamental package for scientific computing with Python

LATEST RELEASE: NUMPY 1.26. VIEW ALL RELEASES

NumPy 2.0 release date: June 16 2024-05-23

Powerful N-dimensional arrays

Fast and versatile, the NumPy vectorization, indexing, and broadcasting concepts are the de-facto standards of array computing today.

Numerical computing tools

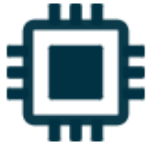
NumPy offers comprehensive mathematical functions, random number generators, linear algebra routines, Fourier transforms, and more.

Open source

Distributed under a liberal [BSD license](#), NumPy is developed and maintained [publicly on GitHub](#) by a vibrant, responsive, and diverse [community](#).

Nearly every scientist working in Python draws on the power of NumPy. NumPy brings the computational power of languages like C and Fortran to Python, a language much easier to learn and use. With this power comes simplicity: a solution in NumPy is often clear and elegant.

Quantum Computing



[QuTiP](#)
[PyQuil](#)
[Qiskit](#)
[PennyLane](#)

Statistical Computing



[Pandas](#)
[statsmodels](#)
[Xarray](#)
[Seaborn](#)

Signal Processing



[SciPy](#)
[PyWavelets](#)
[python-control](#)
[HyperSpy](#)

Image Processing



[Scikit-image](#)
[OpenCV](#)
[Mahotas](#)

Graphs and Networks



[NetworkX](#)
[graph-tool](#)
[igraph](#)
[PyGSP](#)

Astronomy



[AstroPy](#)
[SunPy](#)
[SpacePy](#)

Cognitive Psychology



[PsychoPy](#)

Bioinformatics



[BioPython](#)
[Scikit-Bio](#)
[PyEnsembl](#)
[ETE](#)

Bayesian Inference



[PyStan](#)
[PyMC3](#)
[ArviZ](#)
[emcee](#)

Mathematical Analysis



[SciPy](#)
[SymPy](#)
[cvxpy](#)
[FEniCS](#)

Chemistry



[Cantera](#)
[MDAnalysis](#)
[RDKit](#)
[PyBaMM](#)

Geoscience



[Pangeo](#)
[Simpeg](#)
[ObsPy](#)
[Fatiando a Terra](#)

Geographic Processing



[Shapely](#)
[GeoPandas](#)
[Folium](#)

Architecture & Engineering



[COMPAS](#)
[City_Energy_Analyst](#)
[Sverchok](#)

BASH (LINUX)



PRZECHOWYWANIE I PRZETWARZANIE DANYCH

DLACZEGO LINUX?



```
total 44G
-rw-rw---- 1 szyda upwroclaw 383M Jan 16 2014 lane1_NoIndex_L001_R1_029.fastq.gz
-rw-rw---- 1 szyda upwroclaw 376M Jan 16 2014 lane1_NoIndex_L001_R2_055.fastq.gz
-rw-rw---- 1 szyda upwroclaw 385M Jan 16 2014 lane1_NoIndex_L001_R2_038.fastq.gz
-rw-rw---- 1 szyda upwroclaw 375M Jan 16 2014 lane1_NoIndex_L001_R1_046.fastq.gz
-rw-rw---- 1 szyda upwroclaw 387M Jan 16 2014 lane1_NoIndex_L001_R1_059.fastq.gz
-rw-rw---- 1 szyda upwroclaw 383M Jan 16 2014 lane1_NoIndex_L001_R2_048.fastq.gz
-rw-rw---- 1 szyda upwroclaw 385M Jan 16 2014 lane1_NoIndex_L001_R1_010.fastq.gz
-rw-rw---- 1 szyda upwroclaw 380M Jan 16 2014 lane1_NoIndex_L001_R2_050.fastq.gz
-rw-rw---- 1 szyda upwroclaw 378M Jan 16 2014 lane1_NoIndex_L001_R1_057.fastq.qz
```

surowe

dane
8 TB

pliki pośrednie
7 TB

pliki końcowe
50 GB

+ dane dodatkowe

PRZECHOWYWANIE I PRZETWARZANIE DANYCH

DLACZEGO LINUX?

przysrównanie; 8 rdzeni; 25 GB (genom)

→ ~20 GB

obliczenia równoległe- przyspieszenie obliczeń

→ max 24 rdzenie

czas analiz wszystkich genomów

→ dni/tygodnie ?

System Requirements

1. Hardware:

- a) 64-bit x86-64 CPUs with SSE instructions.
- b) 8 GB main memory (for a genome as large as human's).
- c) 8 GB hard disk (for a genome as large as human's).

2. Software:

- a) 64-bit Linux system



Short Oligonucleotide
Analysis Package

System Requirement

SOAPdenovo aims for large plant and animal genomes, although it also works well on bacteria and fungi genomes. It runs on 64-bit Linux system with a minimum of 5G physical memory. For big genomes like human, about 150 GB memory would be required.

PRZECIĄGNIĘCIE I PRZETWARZANIE DANYCH

DLACZEGO LINUX? DLACZEGO BASH?

Praca na *klastrach obliczeniowych*

- Brak GUI
- W praktyce przetwarzanie i przechowywanie danych biologicznych nie byłoby możliwe bez komputerów o dużej mocy obliczeniowej i o dużych zasobach pamięci dyskowej

Środowisko programistyczne:

- Szybkie przetwarzanie danych
- Pisanie własnych skryptów
- Implementacja programów
- Wiele programów dostępnych jedynie pod Linuxa

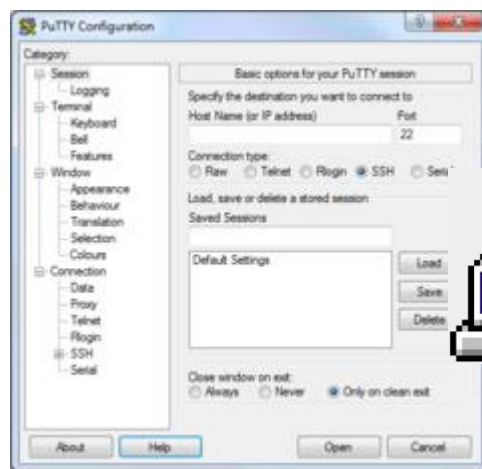


PRZECHOWYWANIE I PRZETWARZANIE DANYCH

DLACZEGO BASH?

Orzeł/Eagle z systemem pamięci masowej XtremIO

2016-11-09



Przy budowie jednego z najszybszych superkomputerów w Polsce – Orzeł/Eagle – zbudowanego przez PCSS, wykorzystano rozwiązania XtremIO firmy Dell EMC. System pamięci masowych firmy Dell EMC, XtremIO, który od początku był projektowany w architekturze All-flash i dzięki temu charakteryzuje się bardzo wysoką wydajnością i opóźnieniem dostępu do danych poniżej 1 ms, umożliwił uzyskanie bardzo wysokich wydajności dzięki ekstremalnie szybkiej obsłudze tzw. meta danych zawierających m.in. informacje o rzeczywistym położeniu danych użytkownika na nośnikach pamięci masowych.

PODSTAWOWE DEFINICJE

Linux – system operacyjny, które oferuje kompletne środowisko programistyczne

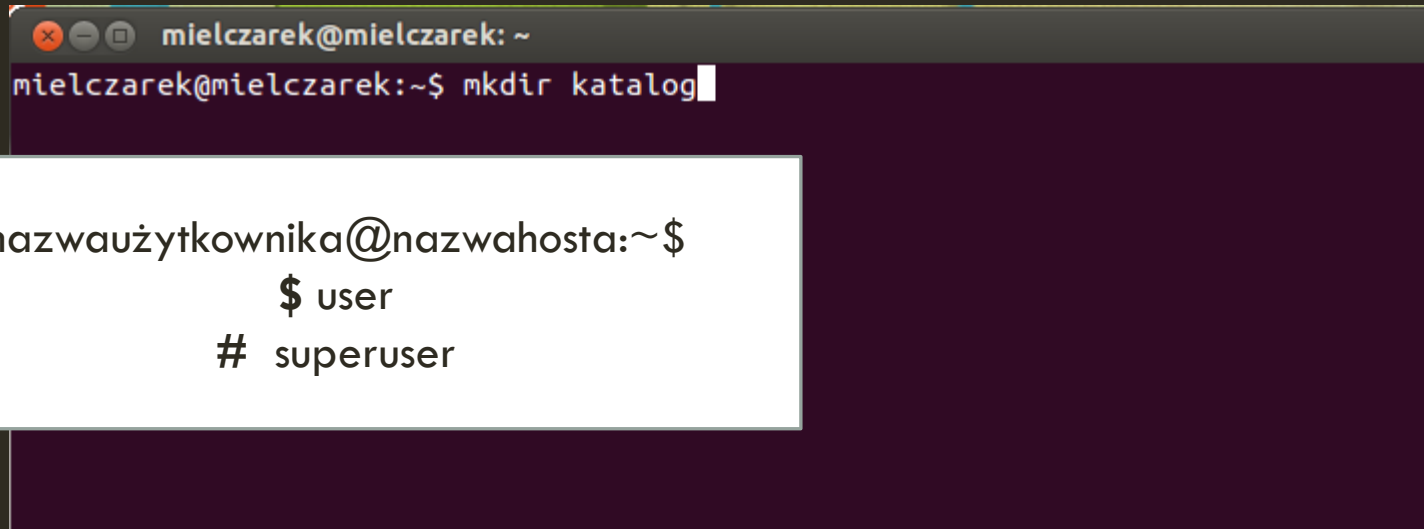
Powłoka – interfejs wiersza poleceń zapewniający komunikację z systemem operacyjnym oraz jego kontrolę

Bash – powłoka (środowisko domyślne)

Skrypt powłoki – plik tekstowy zawierający sekwencje poleceń

Katalog domowy - folder domowy użytkownika; zawiera pliki określające ustawienia każdego użytkownika, przeznaczony na zapisywanie danych, np. dokumentów, obrazów, itp.; standardowo /home/nazwa_użytkownika

TERMINAL



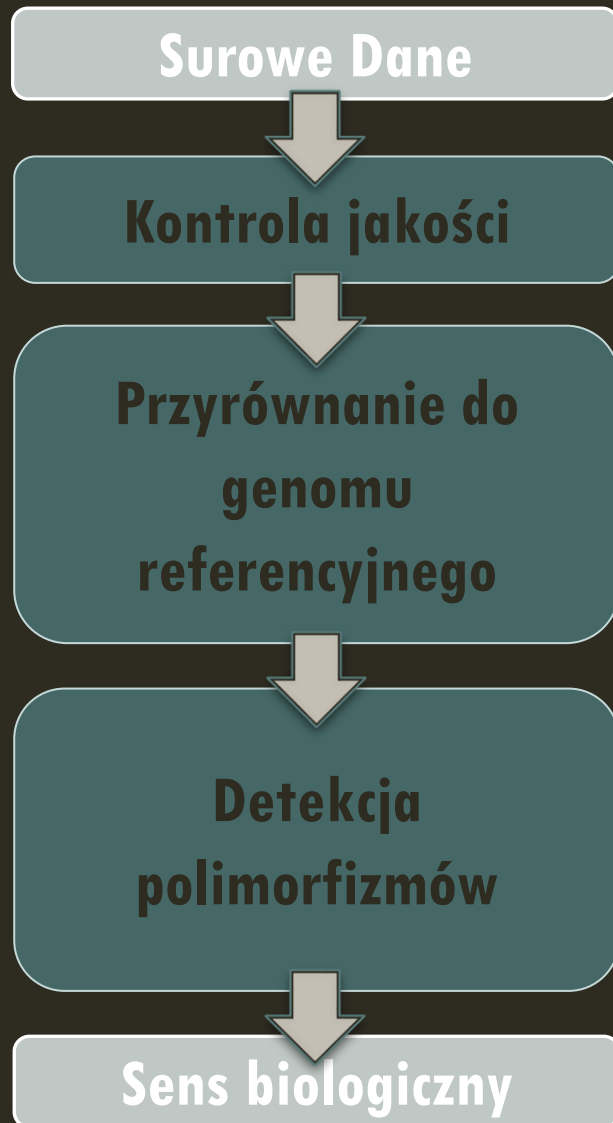
The image shows a terminal window with a dark purple background. The title bar at the top reads 'mielczarek@mielczarek: ~'. The terminal content shows the command 'mielczarek@mielczarek:~\$ mkdir katalog' followed by a cursor. Overlaid on the terminal is a white box containing a list of shell prompts: 'nazważytkownika@nazwahosta:~\$', '\$ user', and '# superuser'.

```
mielczarek@mielczarek: ~  
mielczarek@mielczarek:~$ mkdir katalog
```

nazważytkownika@nazwahosta:~\$
\$ user
superuser

- Polecenia są wpisywane i wykonywane w terminalu powłoki
- Terminal – interaktywny program narzędziowy umożliwiający komunikację użytkownika ze środowiskiem powłoki
- Po otwarciu okna dostępny jest wiersz poleceń

PROBLEM 3: IMPLEMENTACJA PROGRAMÓW



Pipeline = łańcuch przetwarzania danych

Automatyzacja działania wielu programów np. do analizy danych NGS

NISKA JAKOŚĆ I FILTORWANIE DANYCH NGS

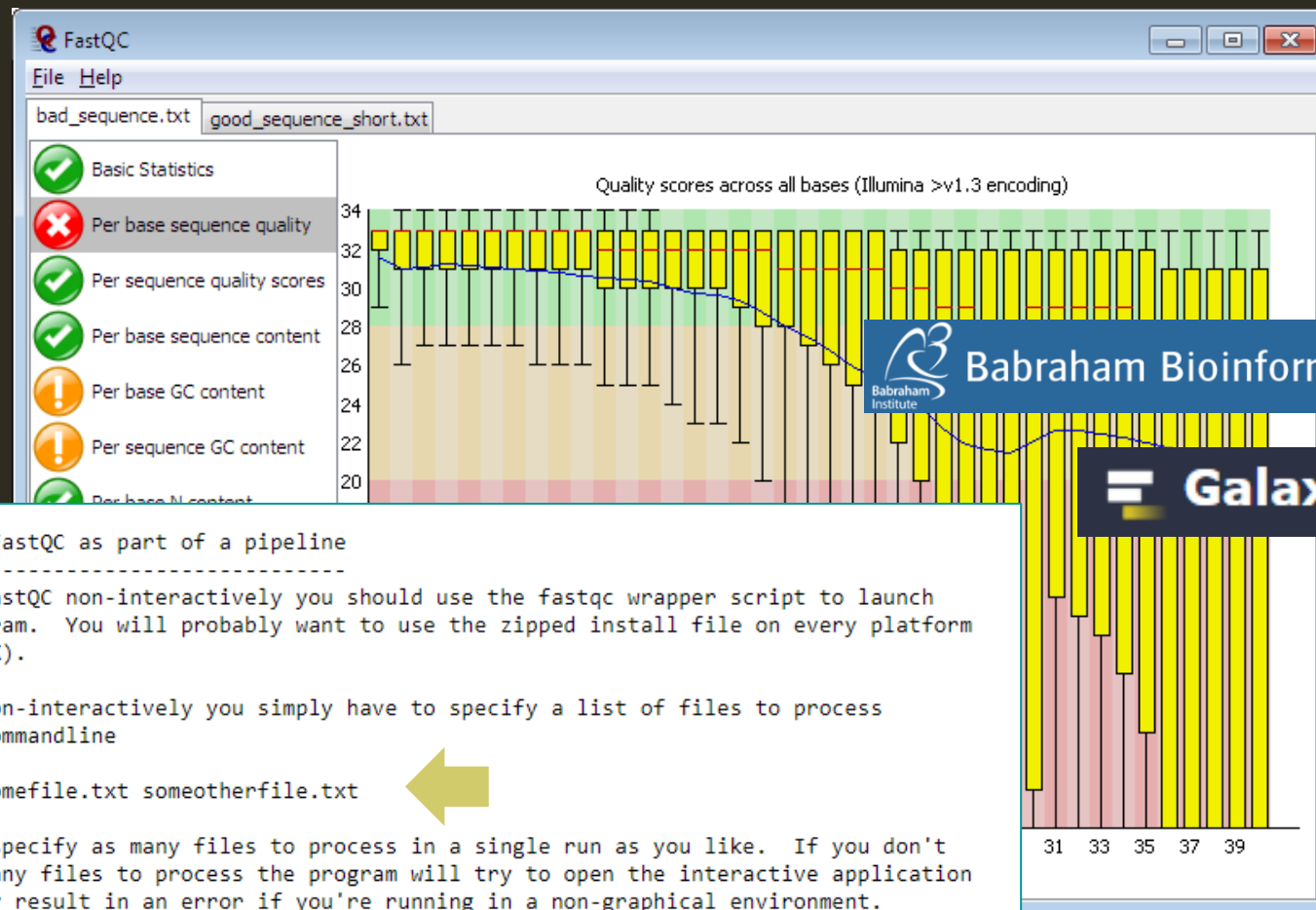
Surowe Dane

Kontrola jakości

Przyrównanie do
genomu
referencyjnego

Detekcja
polimorfizmów

Sens biologiczny



NISKA JAKOŚĆ I FILTROWANIE DANYCH NGS

Surowe Dane

Kontrola jakości

Przyrównanie do
genomu
referencyjnego

Detekcja
polimorfizmów

Sens biologiczny



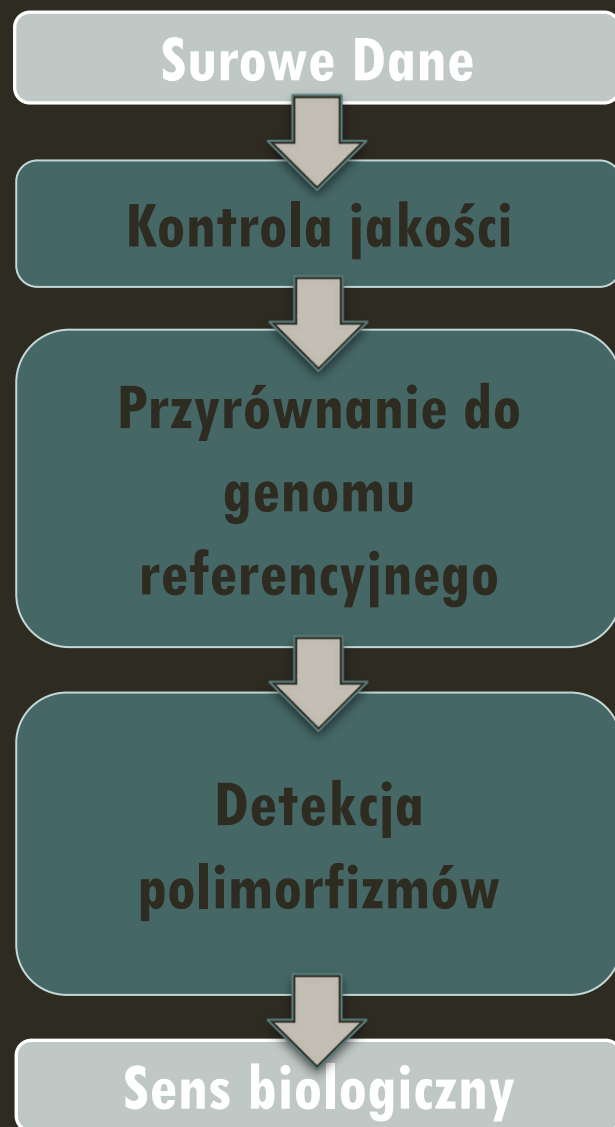
mielczarek@eagle.man.poznan.pl: ~/EAAP_2018/programs/fastqc

```
#!/bin/bash
#SBATCH --ntasks-per-node=6
#SBATCH --time=48:00:00
#SBATCH --mem=4G

fastqc=/home/users/mielczarek/FastQC_v0.11.7/fastqc
files=/home/users/mielczarek/EAAP_2018/data/PRJNA403969/RNA/trimmed/RNA_list.txt
out=/home/users/mielczarek/RNA_data/PRJNA417068_study/out_files/fastqc_out/

$fastqc -t 6 $files -o $out
```


NISKA JAKOŚĆ I FILTORWANIE DANYCH NGS



Trimmomatic: A flexible read trimming tool for Illumina NGS data

Citations

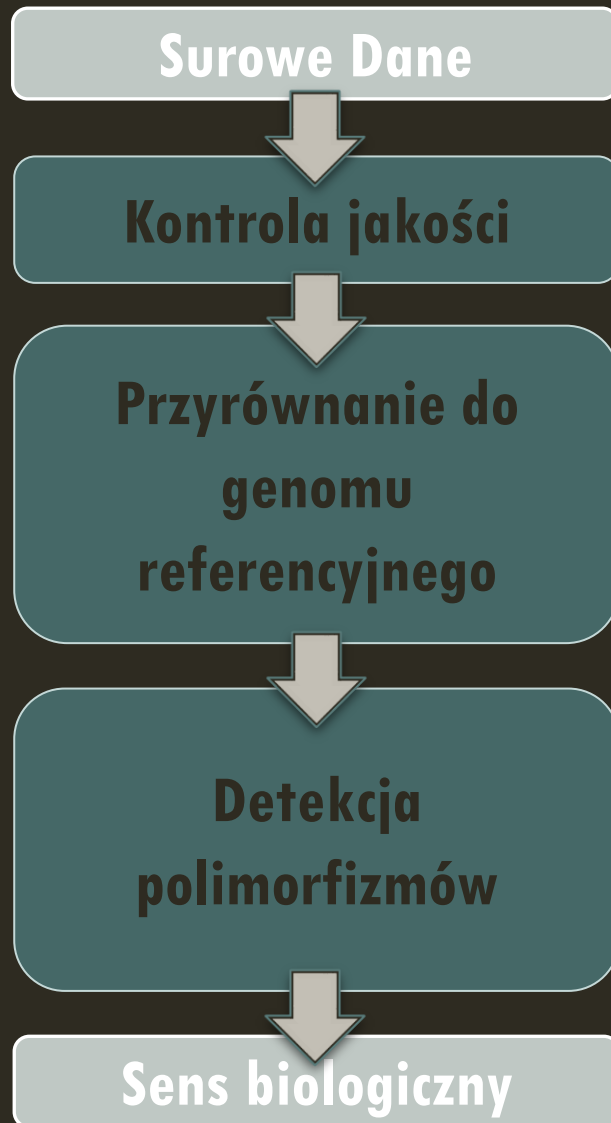
Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina Sequence Data. *Bioinformatics*, btu170.

Trimmomatic performs a variety of useful trimming tasks for illumina paired-end and single ended data. The selection of trimming steps and their associated parameters are supplied on the command line.

The current trimming steps are:

- ILLUMINACLIP: Cut adapter and other illumina-specific sequences from the read.
- SLIDINGWINDOW: Perform a sliding window trimming, cutting once the average quality within the window falls below a threshold.
- LEADING: Cut bases off the start of a read, if below a threshold quality
- TRAILING: Cut bases off the end of a read, if below a threshold quality
- CROP: Cut the read to a specified length
- HEADCROP: Cut the specified number of bases from the start of the read
- MINLEN: Drop the read if it is below a specified length
- TOPHRED33: Convert quality scores to Phred-33
- TOPHRED64: Convert quality scores to Phred-64

NISKA JAKOŚĆ I FILTROWANIE DANYCH NGS



Trimmomatic: A flexible read trimming tool for Illumina NGS data

Citations

Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina Sequence Data. *Bioinformatics*, btu170.

Paired End:

```
java -jar trimmomatic-0.35.jar PE -phred33 input_forward.fq.gz input_reverse.fq.gz  
output_forward_paired.fq.gz output_forward_unpaired.fq.gz output_reverse_paired.fq.gz  
output_reverse_unpaired.fq.gz ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 LEADING:3 TRAILING:3  
SLIDINGWINDOW:4:15 MINLEN:36
```

This will perform the following:

- Remove adapters (ILLUMINACLIP:TruSeq3-PE.fa:2:30:10)
- Remove leading low quality or N bases (below quality 3) (LEADING:3)
- Remove trailing low quality or N bases (below quality 3) (TRAILING:3)
- Scan the read with a 4-base wide sliding window, cutting when the average quality per base drops below 15 (SLIDINGWINDOW:4:15)
- Drop reads below the 36 bases long (MINLEN:36)

Paired End:

```
java -jar trimmomatic-0.35.jar PE -phred33 input_forward.fq.gz input_reverse.fq.gz
output_forward_paired.fq.gz output_forward_unpaired.fq.gz output_reverse_paired.fq.gz
output_reverse_unpaired.fq.gz ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 LEADING:3 TRAILING:3
SLIDINGWINDOW:4:15 MINLEN:36
```

This will perform the following:

- Remove adapters (ILLUMINACLIP:TruSeq3-PE.fa:2:30:10)
- Remove leading low quality or N bases (below quality 3) (LEADING:3)
- Remove trailing low quality or N bases (below quality 3) (TRAILING:3)
- Scan the read with a 4-base wide sliding window, cutting when the average quality per base drops below 15 (SLIDINGWINDOW:4:15)
- Drop reads below the 36 bases long (MINLEN:36)

```
lane4_NoIndex_L004_R1_034.fastq.gz  
lane4_NoIndex_L004_R2_012.fastq.gz  
lane4_NoIndex_L004_R2_026.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
q.gz input_reverse.fq.gz  
output_reverse_paired.fq.gz  
:30:10 LEADING:3 TRAILING:3  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
.fastq.gz  
ality per base drops below 15  
.fastq.gz  
.fastq.gz  
.fastq.gz  
lane4_NoIndex_L004_R1_058.fastq.gz  
lane4_NoIndex_L004_R1_006.fastq.gz  
lane4_NoIndex_L004_R1_002.fastq.gz  
lane4_NoIndex_L004_R2_006.fastq.gz  
lane4_NoIndex_L004_R2_056.fastq.gz  
lane4_NoIndex_L004_R1_027.fastq.qz
```

done