

BUKU PANDUAN
BLAST QUERY OPTIMIZATION
IN SARS-COV-2 PHYLOGENETIC
TREE CONSTRUCTION

disusun oleh
Yazid Zaidan Mujadid

Universitas Gunadarma
Jakarta
2023

DAFTAR ISI

BAB I PERSIAPAN & PERALATAN	5
A. Persiapan Perangkat	6
B. Instalasi Conda	6
C. Menyiapkan lingkungan pengembangan baru pada conda	15
D. Instalasi Jupyter Notebook	17
E. Instalasi Biopython	19
F. Instalasi Matplotlib	19
G. Instalasi MUSCLE5	20
BAB II REPOSITORI PROYEK, FASTA DATASET, DAN NCBI	31
H. Repositori Proyek	32
I. Sars-Cov-2 Wuhan FASTA	33
BAB III IMPLEMENTASI	35
J. BLAST query	36
L. Menyatukan FASTA input dan FASTA Hasil Pencarian	39
M. Ekstrasi Coding Sequence dari FASTA menggunakan biopython ...	39
N. Pemangkasan file FASTA menggunakan BioPython	40
O. Penyelarasan FASTA menggunakan MUSCLE	41
P. Pembentukan pohon filogenetik	42
Q. Melabeli pohon filogenetik dengan Complete FASTA file	44
R. Pencarian Rekursif	46
S. Penggabungan pohon filogenetik	46

DAFTAR GAMBAR

Gambar 1 . Ilustrasi perbandingan lingkungan python biasa dengan menggunakan conda	7
Gambar 2 . Situs web anaconda	8
Gambar 3 . Halaman instalasi miniconda	8
Gambar 4 . Installer miniconda pada sistem operasi Windows, macOS, dan Linux diakses melalui:(https://docs.conda.io/projects/miniconda/en/latest/index.html)	9
Gambar 5 . Installer miniconda pada Windows	11
Gambar 6 . Miniconda license agreement pada Windows	11
Gambar 7 . Tipe instalasi miniconda pada Windows	12
Gambar 8 . Direktori yang digunakan sebagai instalasi miniconda pada Windows	12
Gambar 9 . Opsi tambahan instalasi miniconda pada Windows	13
Gambar 10 . Instalasi miniconda ketika berhasil dipasang pada Windows	14
Gambar 11 . Instalasi miniconda ketika berhasil dipasang pada Windows bag. 2	14
Gambar 12 . Miniconda dapat diakses melalui start menu pada Windows	15
Gambar 13 . Konfirmasi instalasi lingkungan baru pada conda	16
Gambar 14 . Halaman jupyter notebook pada home user	18
Gambar 15 . Halaman jupyter notebook pada spesifik folder	19
Gambar 16 . Bioconda MUSCLE	21
Gambar 17 . Halaman instalasi bioconda pada Windows	22
Gambar 18 . Halaman pengunduhan Muscle5	22
Gambar 19 . Halaman github untuk mengunduh Muscle5	23
Gambar 20 . Konfirmasi pengunduhan MUSCLE5 pada browser Microsoft Edge di Windows 11	23
Gambar 21 . Konfirmasi pengunduhan MUSCLE5 pada browser Microsoft Edge di Windows 11 bag.2	24
Gambar 22 . Direktori instalasi muscle pada Windows	24
Gambar 23 . Start menu windows untuk mengedit environment variables	25
Gambar 24 . Jendela konfigurasi environment variable	26
Gambar 25 . Jendela untuk mengubah konfigurasi variable "Path"	27

Gambar 26 . Jendela pemilihan direktori baru pada variable “Path” menuju folder Muscle	27
Gambar 27 . Jendela variable “Path” setelah direktori Muscle ditambahkan	28
Gambar 28 . Jendela konfigurasi environment variable setelah direktori Muscle ditambahkan pada variable Path	29
Gambar 29 . Program Muscle dapat diakses pada command prompt Windows setelah konfigurasi environment variable selesai	30
Gambar 30 . Repositori proyek penelitian	32
Gambar 31 . Halaman Gene ID dari SARS-CoV-2 NC_045512	33
Gambar 32 . Halaman detail informasi genetik dari virus SARS-CoV-2 ...	34
Gambar 33 . Navigasi pintas untuk menjalankan BLAST menggunakan spesifik halaman sekuens yang sedang dibuka	34
Gambar 34 . Halaman BLAST diarahkan dari detail genetik SARS-CoV-2 .	36
Gambar 35 . Halaman BLAST diarahkan dari detail genetik SARS-CoV-2 .	37
Gambar 36 . Halaman BLAST diarahkan dari detail genetik SARS-CoV-2 .	38
Gambar 37 . Contoh pohon filogenetik pada search result 1	43
Gambar 38 . Pohon filogenetik juga dapat disimpan ke dalam file svg ..	44
Gambar 39 . Pohon filogenetik juga dapat ditampilkan penamaan simpul menggunakan custom function	45

BAB I

PERSIAPAN & PERALATAN

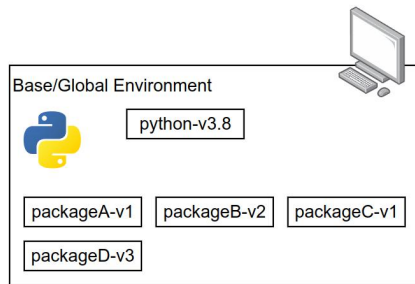
A. Persiapan Perangkat

Berikut ini adalah spesifikasi kebutuhan yang digunakan dalam menjalankan program yang digunakan selama penelitian.

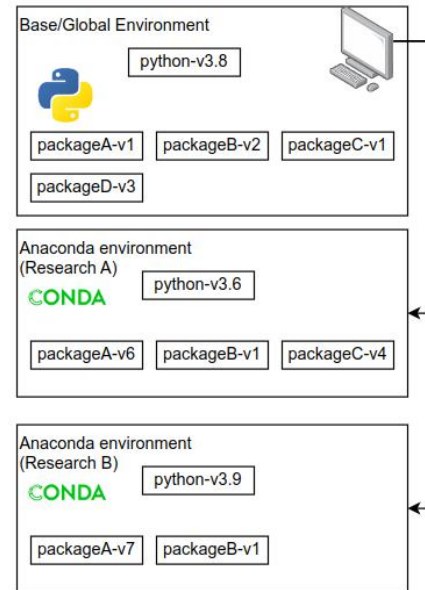
OS	- Windows 10/11 atau - Linux x86_64 kernel 6.4+
CPU	Octa core @ 4.5 Ghz
RAM	8GB
Perangkat Lunak	- Anaconda/miniconda dengan python 3.8 - Jupyter Notebook - Browser untuk membuka antarmuka jupyter - BioPython - Matplotlib - MUSCLE5

B. Instalasi Conda

Conda (conda.io) merupakan salah satu alat yang banyak digunakan untuk tujuan penelitian. Conda memiliki peran penting dalam mengelola lingkungan pustaka dalam manajemen paket pada lingkungan pengembangan Python. Conda dapat digunakan untuk membangun lebih dari satu lingkungan sandbox python dengan set pustaka yang berbeda. Dengan demikian jika ada dua penelitian yang harus menggunakan package yang sama dengan versi yang berbeda dapat dipisah menjadi lingkungan terpisah seperti yang diilustrasikan pada gambar dibawah ini.



(A)



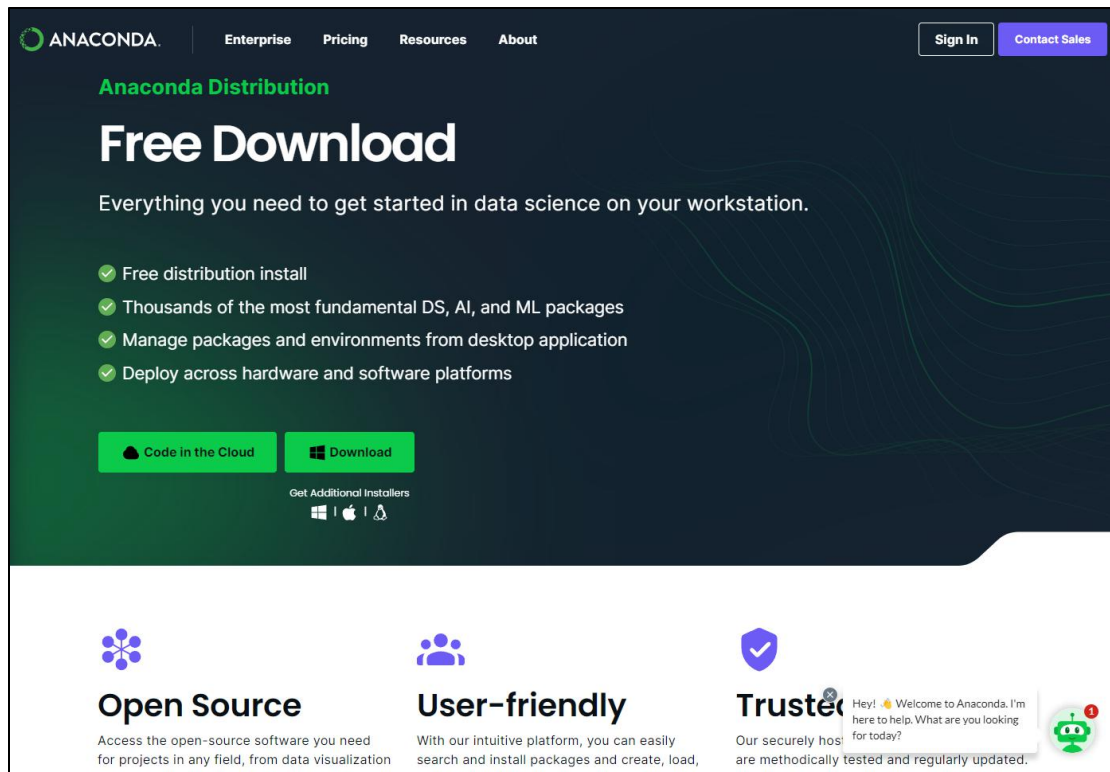
(B)

Gambar 1. Ilustrasi perbandingan lingkungan python biasa dengan menggunakan conda

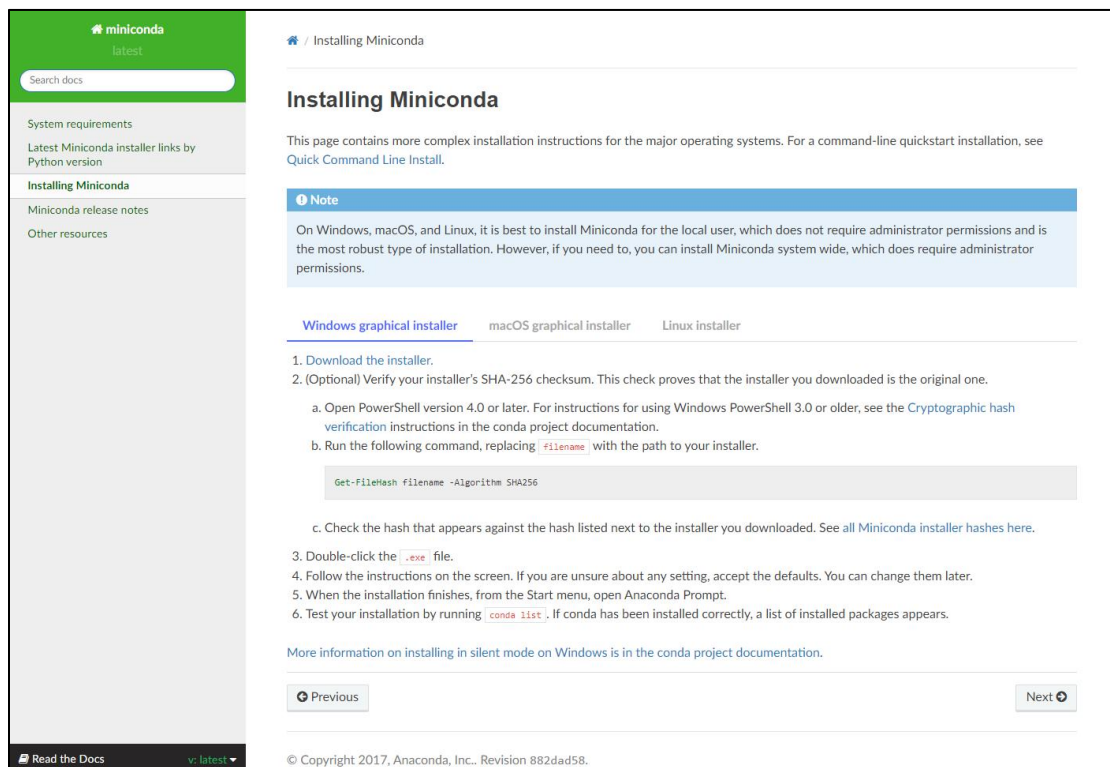
(A) Python dengan pustaka yang bersifat global pada komputer

(B) Komputer dengan bantuan conda memungkinkan untuk menciptakan lingkungan dependency untuk menghindari konflik

Conda sendiri menyediakan dua varian instalasi yaitu anaconda dan miniconda, dimana anaconda sudah dilengkapi dengan berbagai kebutuhan paket disertai antarmuka GUI (Anaconda Navigator) untuk memudahkan penggunaanya dalam mengelola lingkungan pekerjaan, sedangkan miniconda dirancang dengan ukuran yang lebih kecil dan lebih ringan tanpa adanya Anaconda Navigator (Antarmuka GUI) serta hanya memasang package inti dari conda.



Gambar 2. Situs web anaconda



Gambar 3. Halaman instalasi miniconda

miniconda

latest

Search docs

System requirements

Latest Miniconda installer links by Python version

Installing Miniconda

Miniconda release notes

Other resources

Miniconda

Miniconda is a free minimal installer for conda. It is a small bootstrap version of Anaconda that includes only conda, Python, the packages they both depend on, and a small number of other useful packages (like pip, zlib, and a few others). If you need more packages, use the `conda install` command to install from thousands of packages available by default in Anaconda's public repo, or from other channels, like conda-forge or bioconda.

Is Miniconda the right conda install for you? The [Anaconda](#) or [Miniconda](#) page lists some reasons why you might want one installation over the other.

- System requirements
- Latest Miniconda installer links by Python version
- Installing Miniconda
- Miniconda release notes
- Other resources

Latest Miniconda installer links

This list of installers is for the latest release of Python: 3.11.4. For installers for older versions of Python, see [Other installer links](#). For an archive of Miniconda versions, see <https://repo.anaconda.com/miniconda/>.

Latest - Conda 23.5.2 Python 3.11.4 released July 13, 2023

Platform	Name	SHA256 hash
Windows	Miniconda3 Windows 64-bit	00e837854233686204c798a83966f107344a8ad554076600efebc23f40e2914
macOS	Miniconda3 macOS Intel x86 64-bit bash	1622e7a9f960e7d3d892c2d8153054c06ffe3e6b979d9313280e56d5258164b
	Miniconda3 macOS Intel x86 64-bit pkg	2236a243b6cbe9f16ec324ecc9651102494c03d541791d4461200b6a781a0b4
	Miniconda3 macOS Apple M1 64-bit bash	c8f436dbd130f171d390d7b4fc8669c223f1380a77890839598dc1611a35644
	Miniconda3 macOS Apple M1 64-bit pkg	837371f3b6e8ae2d65bdfc8370e6be81205c4ff9f40bc04e0022f840f9b4fe5
Linux	Miniconda3 Linux 64-bit	634076d95e409c44ade4085552b97ebc706d49245ed1a8300220b0406de5817
	Miniconda3 Linux-aarch64 64-bit	3962738cfac270ee4ff30d90e382aecf6b330581206401964577470157749878
	Miniconda3 Linux-ppc64le 64-bit	92237c284430d15000ec004f2f7440145e02c05513a00993c2f191e043d1b29
	Miniconda3 Linux-s390x 64-bit	22184cd7f899275c3263ef987f83785746d6884f4300b05d1fe5733ca770550

Gambar 4. Installer miniconda pada sistem operasi Windows, macOS, dan Linux diakses melalui:
(<https://docs.conda.io/projects/miniconda/en/latest/index.html>)

Menginstal pada sistem operasi Linux

Dari laman instalasi miniconda menyediakan cara instalasi cepat pada sistem operasi linux dengan pertama membuat direktori miniconda3 pada home user melalui perintah berikut.

```
terminal/command prompt
mkdir -p ~/miniconda3
```

Kemudian pada direktori tersebut dapat diunduh installer miniconda terbaru melalui perintah “wget” seperti dibawah ini.

```
terminal/command prompt
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh -O ~/miniconda3/miniconda.sh
```

Setelah installer script berhasil diunduh, maka file tersebut dapat dieksekusi melalui terminal untuk memuat paket dan file binary ke dalam direktori yang sama.

terminal/command prompt
<code>bash ~/miniconda3/miniconda.sh -b -u -p ~/miniconda3</code>

Perintah dibawah ini opsional, untuk menghapus shell script instalasi yang diunduh diawal.

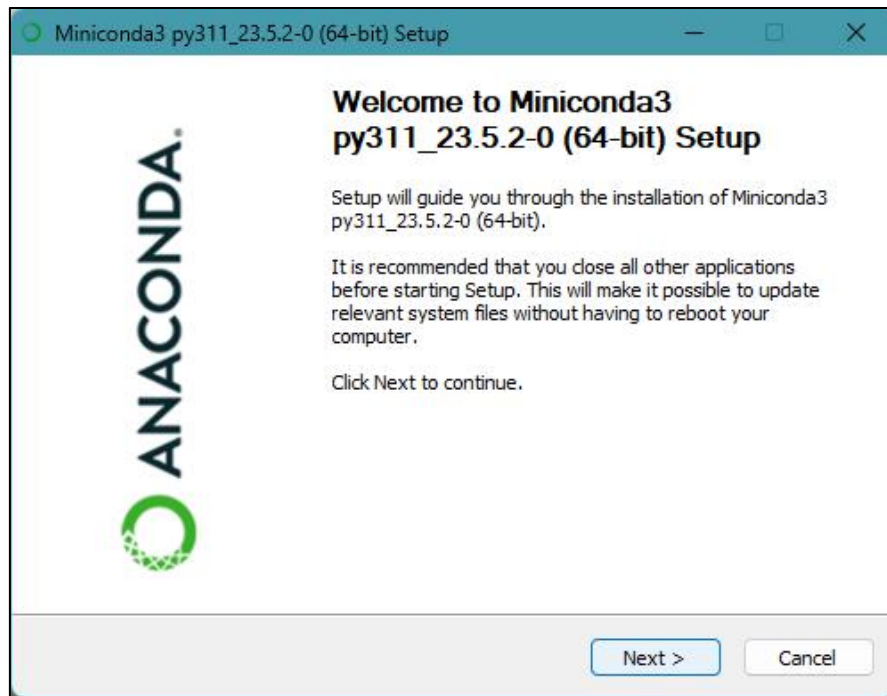
terminal/command prompt
<code>rm -rf ~/miniconda3/miniconda.sh</code>

Terakhir, miniconda perlu diinisialisasi terlebih dahulu supaya perintahnya dapat diakses melalui shell command sesuai dengan jenis shell yang digunakan pada sistem oprasi linux. Sebagai contoh, perintah di bawah ini akan mengkonfigurasi environment variabel pada bash shell dan zsh shell.

terminal/command prompt
<code>~/miniconda3/bin/conda init bash</code>
<code>~/miniconda3/bin/conda init zsh</code>

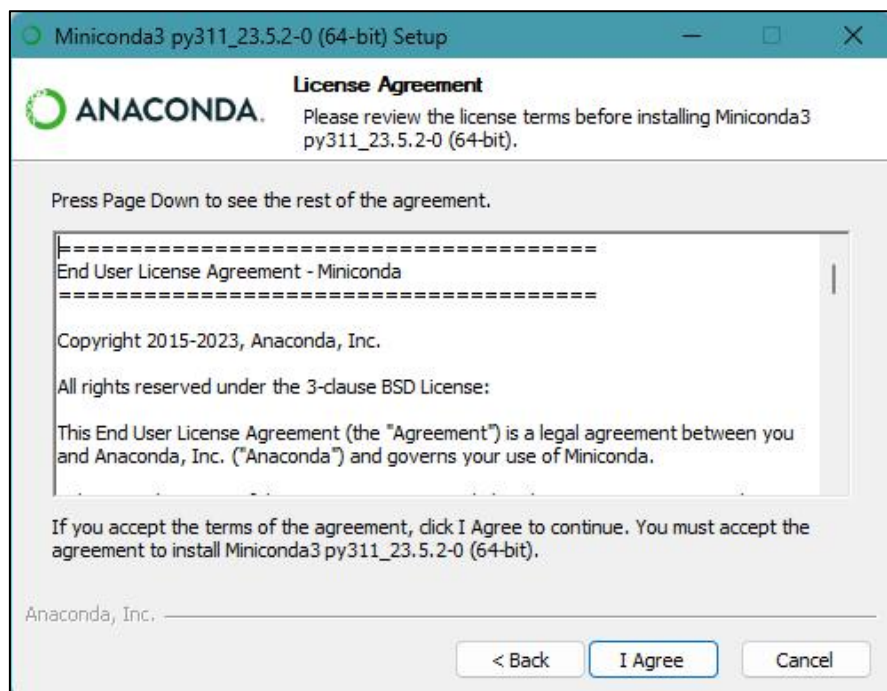
Menginstal pada sistem operasi Windows

Untuk sistem operasi windows dapat membuka halaman ini <https://docs.conda.io/projects/miniconda/en/latest/index.html> untuk kemudian mengunduh GUI installer dari miniconda.

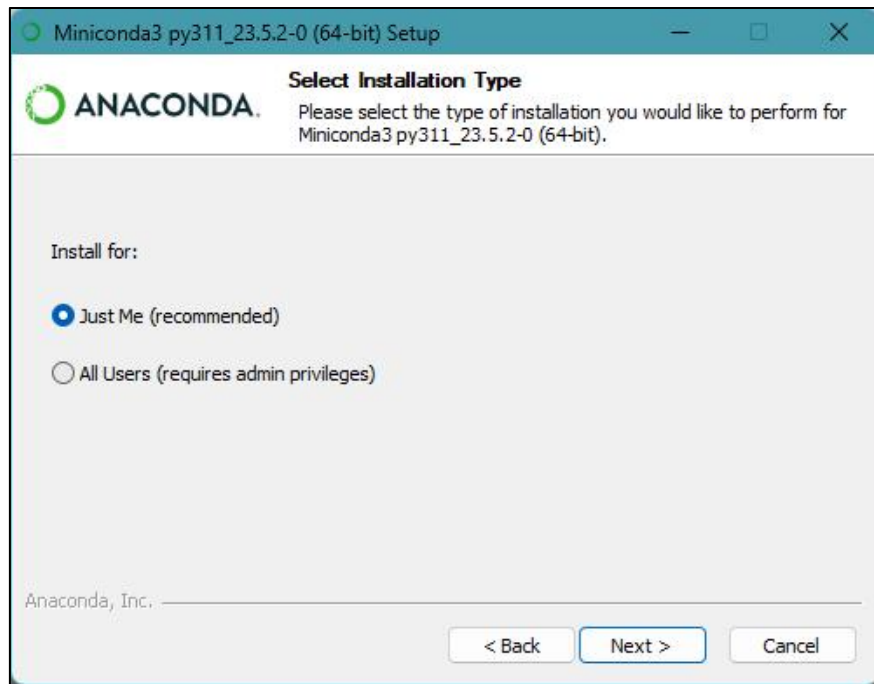


Gambar 5. Installer miniconda pada Windows

Pada installer setup pengguna perlu menyetujui *license agreement* terlebih dahulu sebelum memasang miniconda seperti gambar di bawah.

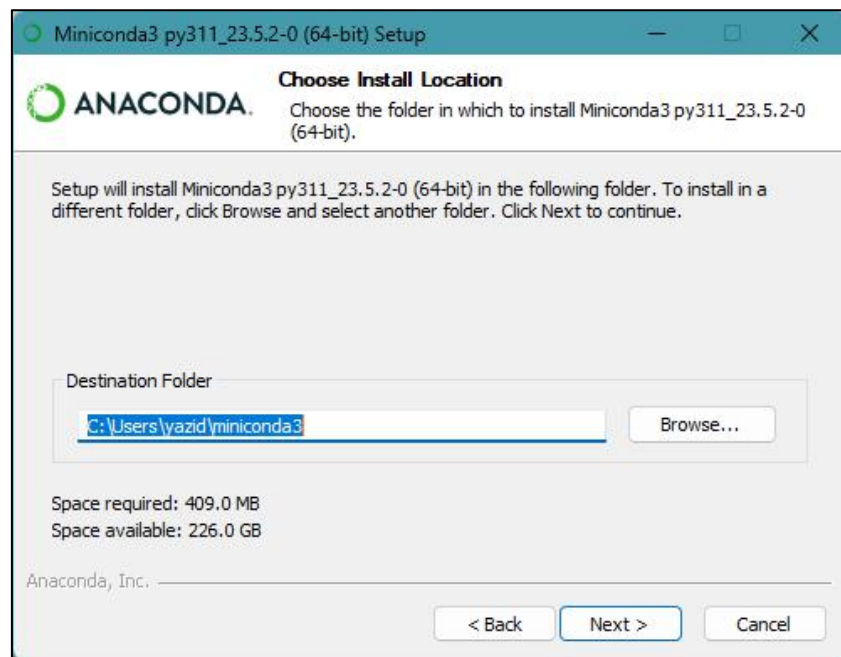


Gambar 6. Miniconda license agreement pada Windows



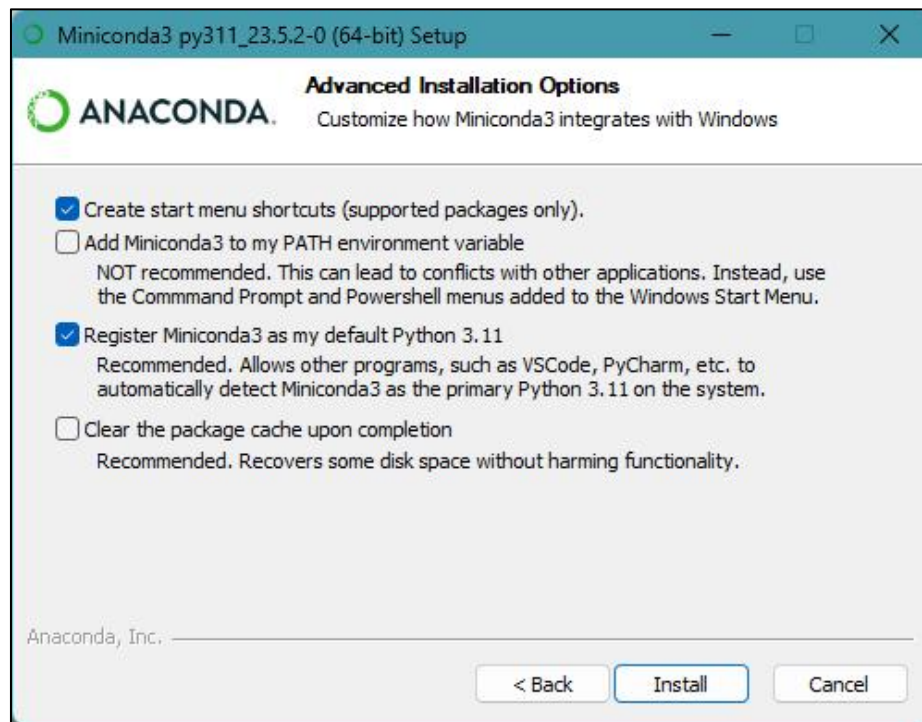
Gambar 7. Tipe instalasi miniconda pada Windows

Secara default, miniconda merekomendasikan instalasi dipasang hanya pada user profile yang sedang memiliki sesi pada sistem operasi windows.



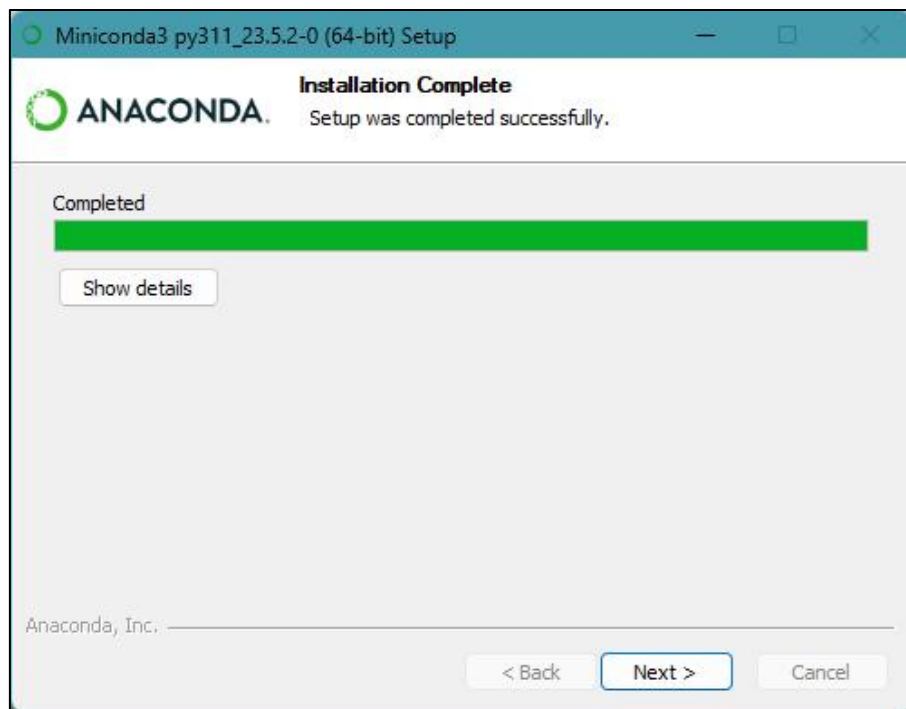
Gambar 8. Direktori yang digunakan sebagai instalasi miniconda pada Windows

Kemudian pada gambar 8, pengguna perlu menentukan direktori yang akan digunakan sebagai tempat instalasi miniconda, dimana secara default akan dipasang pada folder miniconda3 tepat pada folder home user (serupa dengan konfigurasi default pada sistem operasi linux).

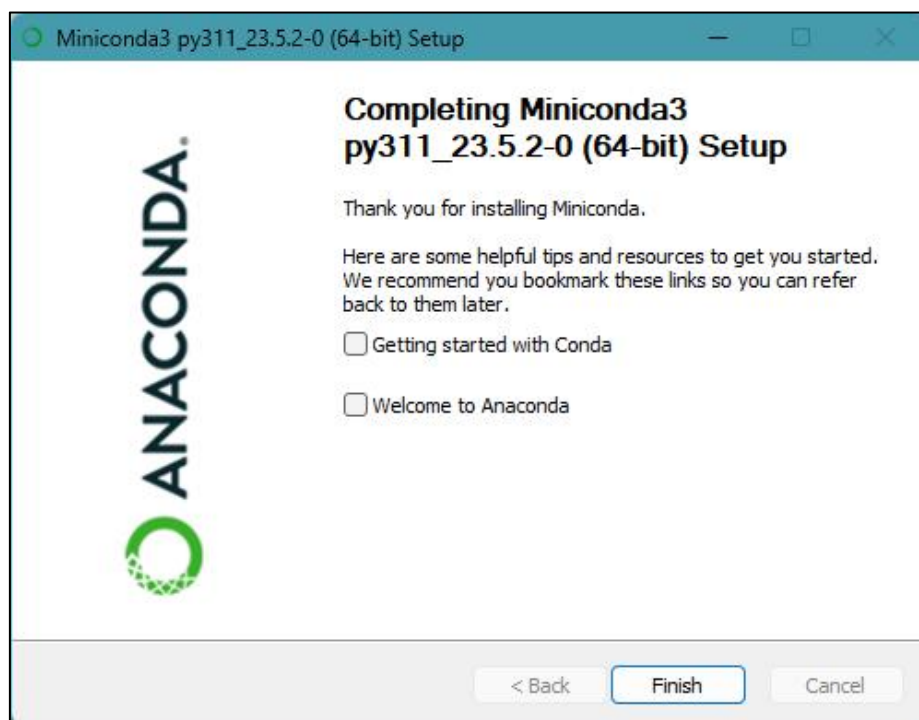


Gambar 9. Opsi tambahan instalasi miniconda pada Windows

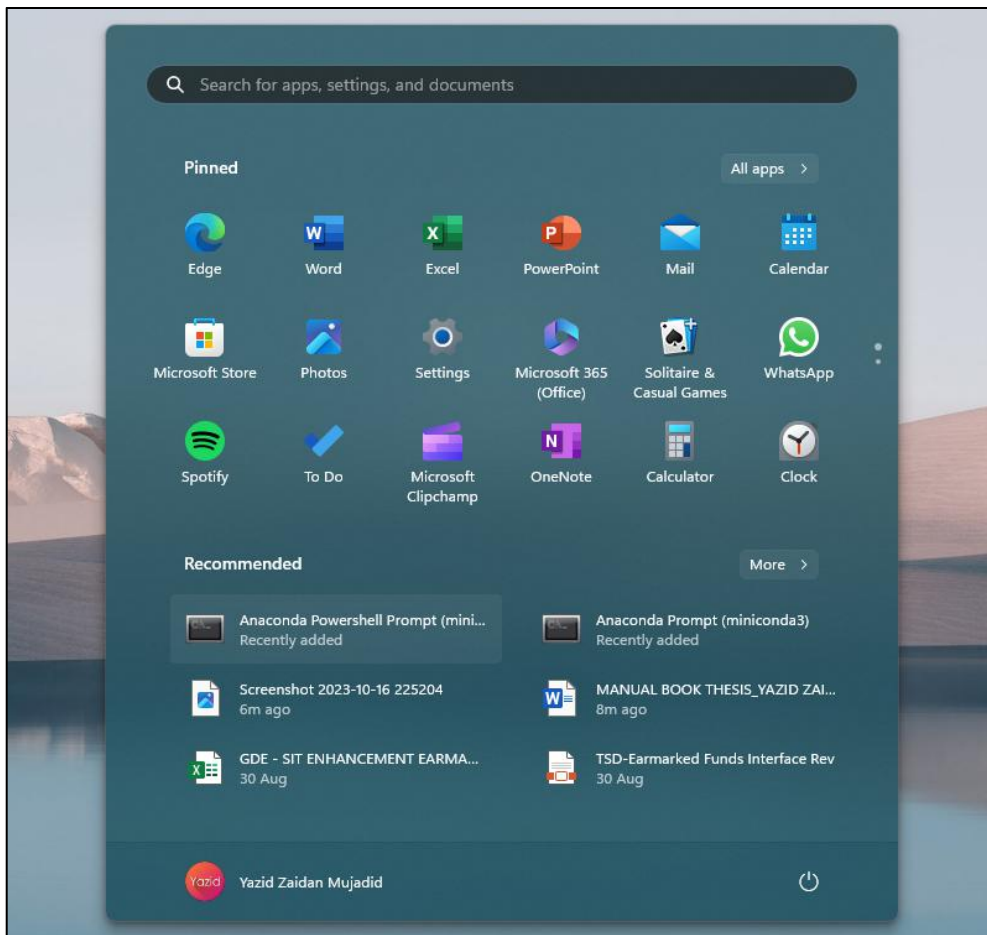
Terakhir opsi instalasi miniconda menyediakan "Add Miniconda3 to my PATH environment variable" supaya perintah miniconda dapat diakses secara langsung melalui command prompt, tetapi opsi ini tidak direkomendasikan pada sistem operasi windows bagi pengguna pemula dan secara default dapat diakses melalui start menu shortcut.



Gambar 10. Instalasi miniconda ketika berhasil dipasang pada Windows



Gambar 11. Instalasi miniconda ketika berhasil dipasang pada Windows bag. 2



Gambar 12. Miniconda dapat diakses melalui start menu pada Windows

Setelah instalasi selesai secara default miniconda dapat diakses melalui start menu shortcut dan tersedia dalam dua jenis yaitu powershell prompt dan command prompt.

C. Menyiapkan lingkungan pengembangan baru pada conda

Untuk menghindari adanya konflik paket pustaka pada sistem operasi ataupun lingkungan lain, dapat dibuat lingkungan pengembangan baru pada program conda menggunakan perintah berikut.

```
terminal/command prompt
conda create -n genome-sequence-research python=3.8
```


Perintah diatas akan membuat lingkungan pengembangan dengan nama “genome-sequence-research” spesifik menggunakan python versi 3.8 (lingkungan ini ditempatkan terpisah dengan lingkungan global jika ada python dengan versi lain yang sudah terpasang pada sistem operasi). Jika perintah di atas dijalankan, maka akan muncul prompt terminal untuk mengkonfirmasi instalasi dengan menampilkan terlebih dahulu daftar package yang akan dipasang dalam membangun lingkungan baru sebagai berikut.

```
The following NEW packages will be INSTALLED:

_libgcc_mutex      conda-forge/linux-64::_libgcc_mutex-0.1-conda_forge
_openmp_mutex      conda-forge/linux-64::_openmp_mutex-4.5-2_gnu
bzip2              conda-forge/linux-64::bzip2-1.0.8-h7f98852_4
ca-certificates    conda-forge/linux-64::ca-certificates-2023.5.7-hbcca054_0
ld_impl_linux-64   conda-forge/linux-64::ld_impl_linux-64-2.40-h41732ed_0
libffi             conda-forge/linux-64::libffi-3.4.2-h7f98852_5
libgcc-ng          conda-forge/linux-64::libgcc-ng-13.1.0-he5830b7_0
libgomp            conda-forge/linux-64::libgomp-13.1.0-he5830b7_0
libns1             conda-forge/linux-64::libns1-2.0.0-h7f98852_0
libsqlite          conda-forge/linux-64::libsqlite-3.42.0-h2797004_0
libuuid           conda-forge/linux-64::libuuid-2.38.1-h0b41bf4_0
libzlib            conda-forge/linux-64::libzlib-1.2.13-hd590300_5
ncurses            conda-forge/linux-64::ncurses-6.4-hcb278e6_0
openssl            conda-forge/linux-64::openssl-3.1.1-hd590300_1
pip                conda-forge/noarch::pip-23.2-pyhd8ed1ab_0
python             conda-forge/linux-64::python-3.8.17-he550d4f_0_cpython
readline           conda-forge/linux-64::readline-8.2-h8228510_1
setuptools         conda-forge/noarch::setuptools-68.0.0-pyhd8ed1ab_0
tk                 conda-forge/linux-64::tk-8.6.12-h27826a3_0
wheel              conda-forge/noarch::wheel-0.40.0-pyhd8ed1ab_1
xz                 conda-forge/linux-64::xz-5.2.6-h166bdaf_0

Proceed ([y]/n)? _
```

Gambar 13. Konfirmasi instalasi lingkungan baru pada conda

Instalasi dapat dieksekusi setelah menerima user keyboard input “y”, yang mana setelah instalasi selesai untuk berpindah dari lingkungan kerja global/base ke lingkungan conda, maka perlu dijalankan perintah “conda activate” dilanjutkan dengan nama lingkungan yang telah disiapkan sebelumnya.

terminal/command prompt
conda activate s2-genome-sequence-research

D. Instalasi Jupyter Notebook

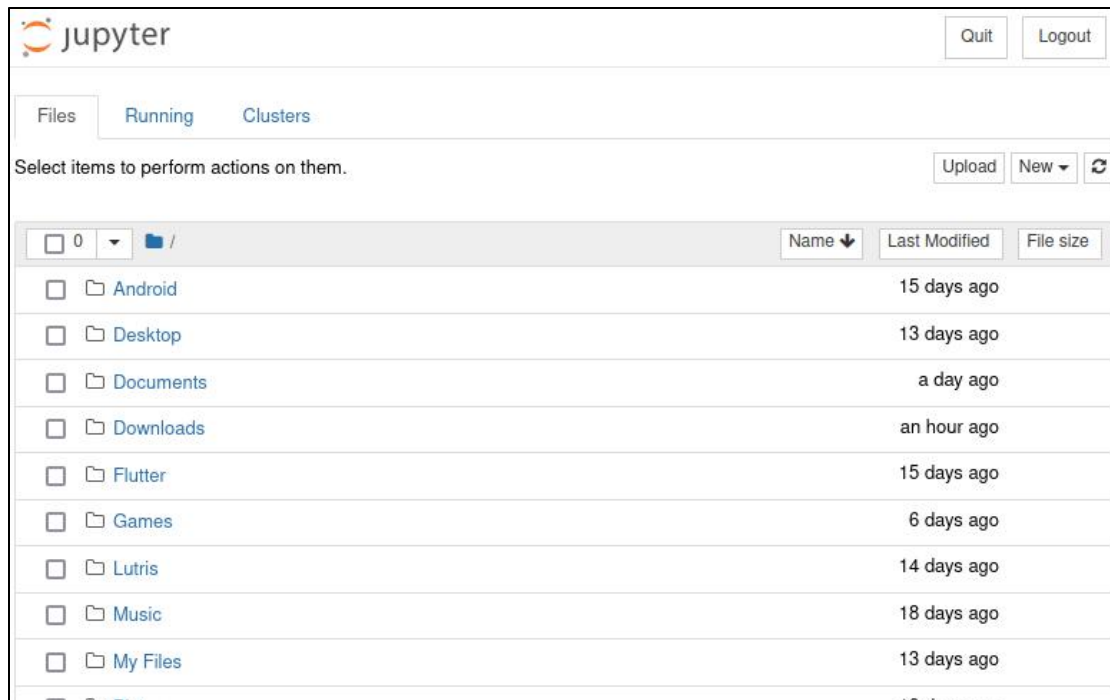
Berbeda dengan pemrograman umum, sesuai namanya (notebook) file dengan format ini mampu untuk mendokumentasikan percobaan pada pemrograman python jauh lebih mudah menggunakan notebook virtualnya. Jupyter notebook mampu mengeksekusi perintah python dan menampilkan outputnya secara langsung pada lembar kerja sekaligus dilengkapi dengan dukungan syntax markdown. Secara antarmuka jupyter notebook dibangun atas susunan blok python syntax (untuk pemrograman) dan markdown (untuk dokumentasi). Menggunakan shortcut (shift + enter) pada blok tersebut sistem otomatis akan mengeksekusi syntax python atau menampilkan teks markdown. Berikut ini adalah perintah untuk memasang program jupyter notebook.

terminal/command prompt
<code>conda install jupyter</code>

Setelah program ini dipasang, jupyter notebook dapat dijalankan menggunakan perintah di bawah ini.

terminal/command prompt
<code>jupyter notebook</code>

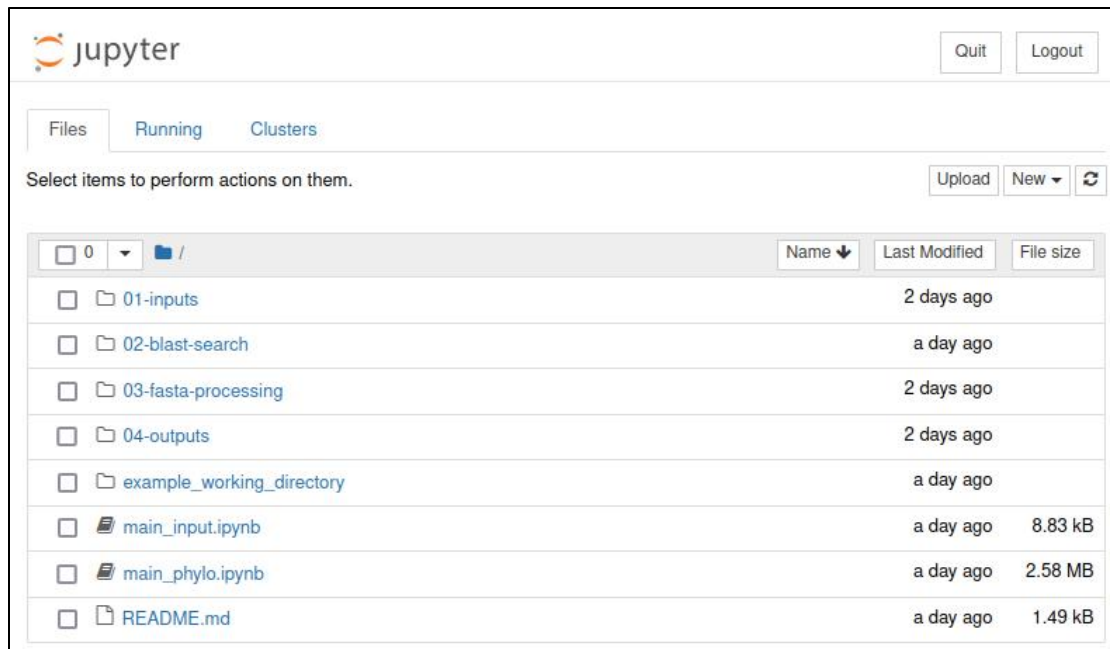
Perintah di atas secara otomatis akan membuka browser default yang dimiliki oleh sistem untuk membuka antarmuka jupyter notebook pada direktori aktif yang sedang digunakan pada terminal/command prompt.



Gambar 14. Halaman jupyter notebook pada home user

Perintah ini juga dapat dijalankan dengan terlebih dahulu mengarahkan terminal/command prompt pada repositori proyek supaya file manager jupyter akan langsung menampilkan direktori tersebut sebagai lembar kerja utama

terminal/command prompt
<code>cd "insert the path to your repository here"</code>
<code>jupyter notebook</code>



Gambar 15. Halaman jupyter notebook pada spesifik folder

E. Instalasi Biopython

BioPython (<https://biopython.org/>) merupakan pustaka utama yang digunakan pada program ini, digunakan untuk mengolah file FASTA. Menggunakan conda package manager pustaka biopython dapat dipasang menggunakan perintah di bawah ini.

```
terminal/command prompt
conda install -c conda-forge biopython
```

Perintah ini akan memasang biopython beserta dependency package yang dibutuhkan sehingga biopython dapat berjalan dengan baik pada lingkungan pengembangan yang sedang diaktifkan.

F. Instalasi Matplotlib

Matplotlib (<https://matplotlib.org/>) merupakan pustaka yang cukup sering digunakan pada lingkungan kerja python notebook dengan peran utamanya dalam memvisualisasikan data yang bersifat statis, dinamis, ataupun interaktif. Matplotlib sendiri

memiliki integrasi yang cukup baik pada python notebook sehingga hanya dengan menjalankan perintah tertentu pengguna dapat memvisualisasikan data yang telah didefinisikan sebelumnya secara cepat. Matplotlib juga mampu menyimpan hasil visualisasi ke dalam file seperti gambar. Pustaka matplotlib sendiri digunakan untuk memvisualisasikan pohon filogenetik yang telah diolah menggunakan BioPython dan MUSCLE. Instalasi matplotlib dapat dilakukan dengan perintah berikut pada lingkungan conda.

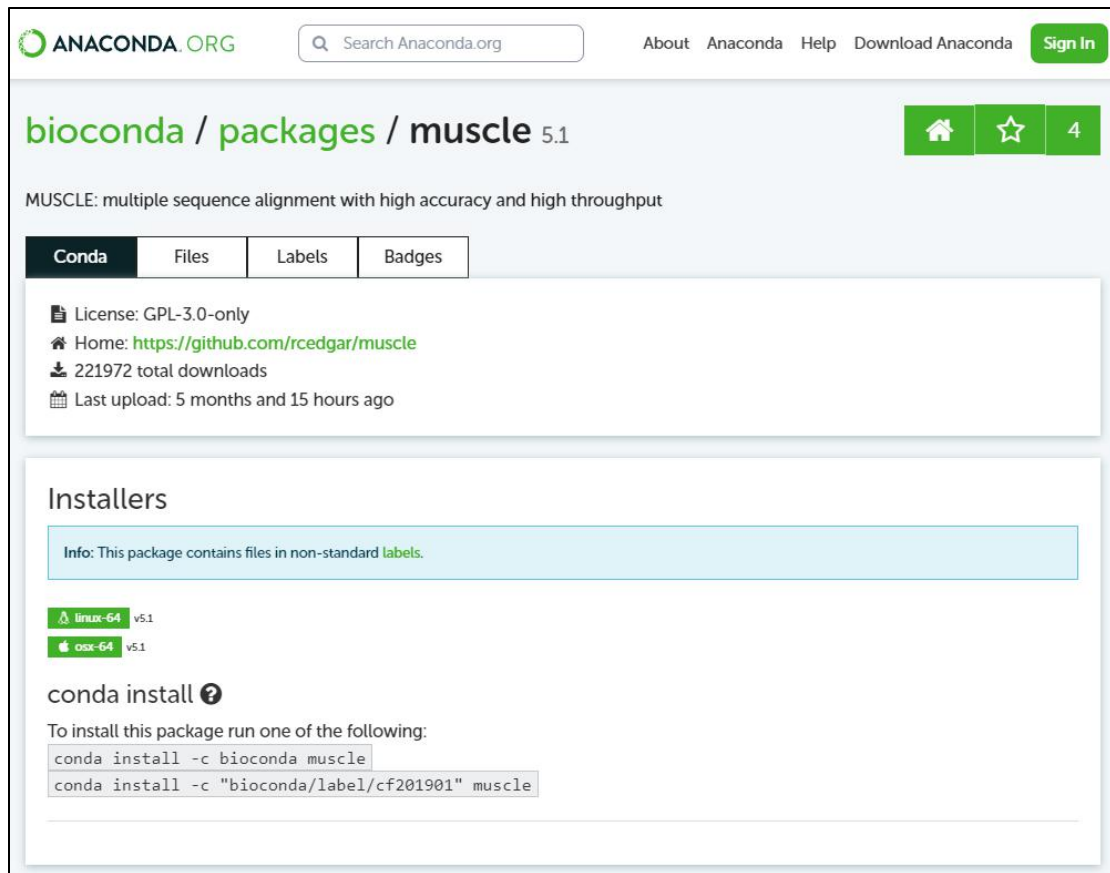
terminal/command prompt
<pre>conda install matplotlib</pre>

G. Instalasi MUSCLE5

Saat ini MUSCLE tersedia dalam dua versi (versi 3 dan versi 5), dimana cara menggunakan dan algoritma yang diterapkan pada kedua versi cukup berbeda. Penelitian ini menggunakan program MUSCLE 5 yang dapat diunduh melalui halaman <https://www.drive5.com/muscle>.

Instalasi MUSCLE5 pada Linux

Bagi pengguna Linux, MUSCLE5 sudah tersedia pada channel bioconda dan dapat diakses langsung instalasinya melalui conda package manager.



ANACONDA.ORG

Search Anaconda.org

About Anaconda Help Download Anaconda Sign In

bioconda / packages / muscle 5.1

MUSCLE: multiple sequence alignment with high accuracy and high throughput

Conda Files Labels Badges

License: GPL-3.0-only
Home: <https://github.com/rcedgar/muscle>
221972 total downloads
Last upload: 5 months and 15 hours ago

Installers

Info: This package contains files in non-standard labels.

linux-64 v5.1
osx-64 v5.1

conda install ?

To install this package run one of the following:

```
conda install -c bioconda muscle  
conda install -c "bioconda/label/cf201901" muscle
```

Gambar 16. Bioconda MUSCLE

Dengan mengaktifkan lingkungan python “genome-sequence-research” dapat dipasang MUSCLE5 dengan perintah berikut.

```
terminal/command prompt  
  
conda install -c bioconda muscle
```

Instalasi MUSCLE5 pada Windows

MUSCLE5 belum tersedia pada Bioconda untuk sistem operasi Windows tetapi masih dapat dipasang secara manual berdasarkan dokumentasi dari drive5.com.

Windows installation

At the time of writing, **MUSCLE** is not available through **Bioconda** for Windows.

1. Open <http://drive5.com/muscle/downloads.htm> with your web browser.
2. Download the latest Windows Intel i86 binary, currently `muscle3.8.31_i86win32.exe`. This will be placed in your `Downloads` directory.
3. In `git bash` change to your home directory with the command `cd`.
4. Create a new directory called `bin` with the command `mkdir bin`.
5. Copy the `MUSCLE` program to this new directory with the command `cp Downloads/muscle3.8.31_i86win32.exe bin/muscle.exe`. This creates a new command called `muscle` which runs the alignment program.
6. Test that the program can be run by executing the command `muscle` in the terminal.

In total, the sequence of commands will be:

```
$ cd
$ mkdir bin
$ cp Downloads/muscle3.8.31_i86win32.exe bin/muscle.exe
$ muscle
```

Gambar 17. Halaman instalasi bioconda pada Windows

[Home](#) [Software](#) [Services](#) [About](#) [Contact](#)

Muscle5

MUSCLE has been cited by
52,365 papers
[Google scholar](#)
Last updated 16 Oct 2023

[Download](#)

[Documentation](#)

[Support and feedback](#)

[MUSCLE v3](#)

Next-generation MUSCLE

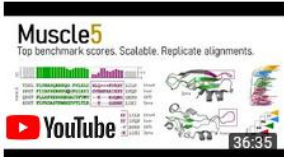
Muscle v5 is a major re-write of MUSCLE based on new algorithms.

Highest accuracy, scalable to thousands of sequences

Compared to previous versions, Muscle v5 is much more accurate, is often faster, and scales to much larger datasets. At the time of writing (late 2021), Muscle v5 has the highest scores on multiple alignment benchmarks including Balibase, Bralibase, Prefab and Balifam. It can align tens of thousands of sequences with high accuracy on a low-cost commodity computer (say, an 8-core Intel CPU with 32 Gb RAM). On large datasets, Muscle v5 is 20-30% more accurate than MAFFT and Clustal-Omega.

Alignment ensembles

Muscle v5 can generate ensembles of high-accuracy alternative alignments. All replicates have equal average accuracy on benchmark test, including the MSA made with default parameters. By comparing results of downstream analysis (trees, structure prediction...) on different replicates, you can assess the effects of alignment errors on your study.



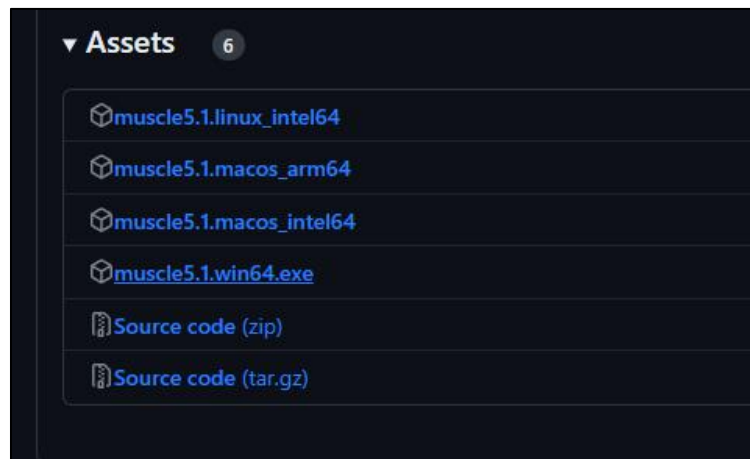
Muscle5
Top benchmark scores. Scalable. Replicate alignments.

YouTube

36:35

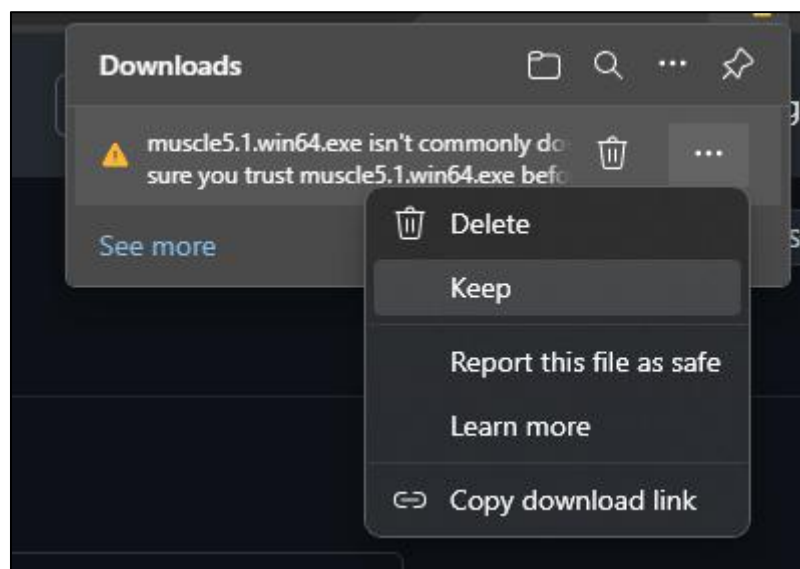
Gambar 18. Halaman pengunduhan Muscle5

Melalui halaman <https://www.drive5.com/muscle/>, akan diarahkan pada halaman github berisi aset berisi instalasi dan source code MUSCLE itu sendiri. Pengguna Windows cukup mengunduh muscle dengan ekstensi .exe.

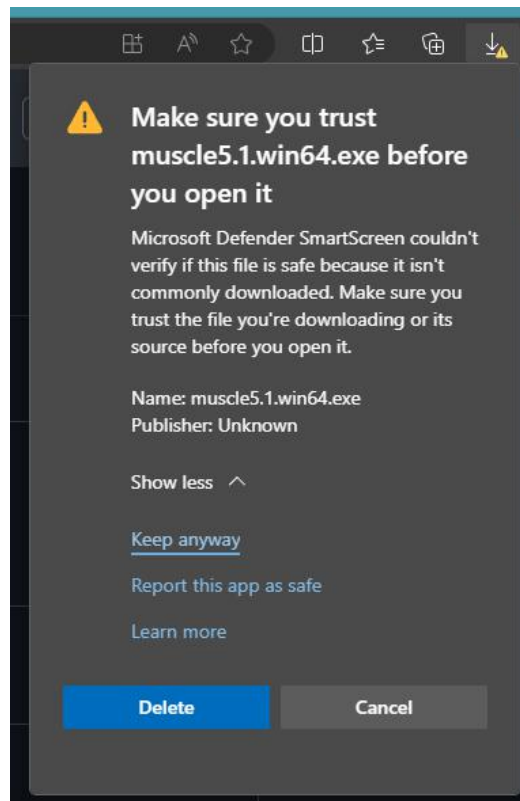


Gambar 19. Halaman github untuk mengunduh Muscle5

Pada beberapa kasus browser instalasi dari source yang tidak dikenal tidak langsung terunduh dan perlu konfirmasi secara manual dari pengguna seperti yang ditunjukkan pada gambar di bawah ini.

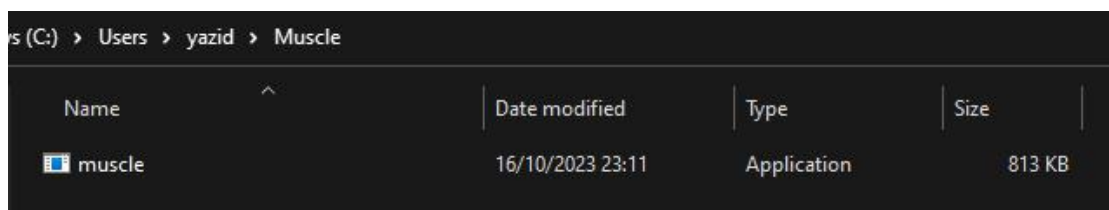


Gambar 20. Konfirmasi pengunduhan MUSCLE5 pada browser Microsoft Edge di Windows 11



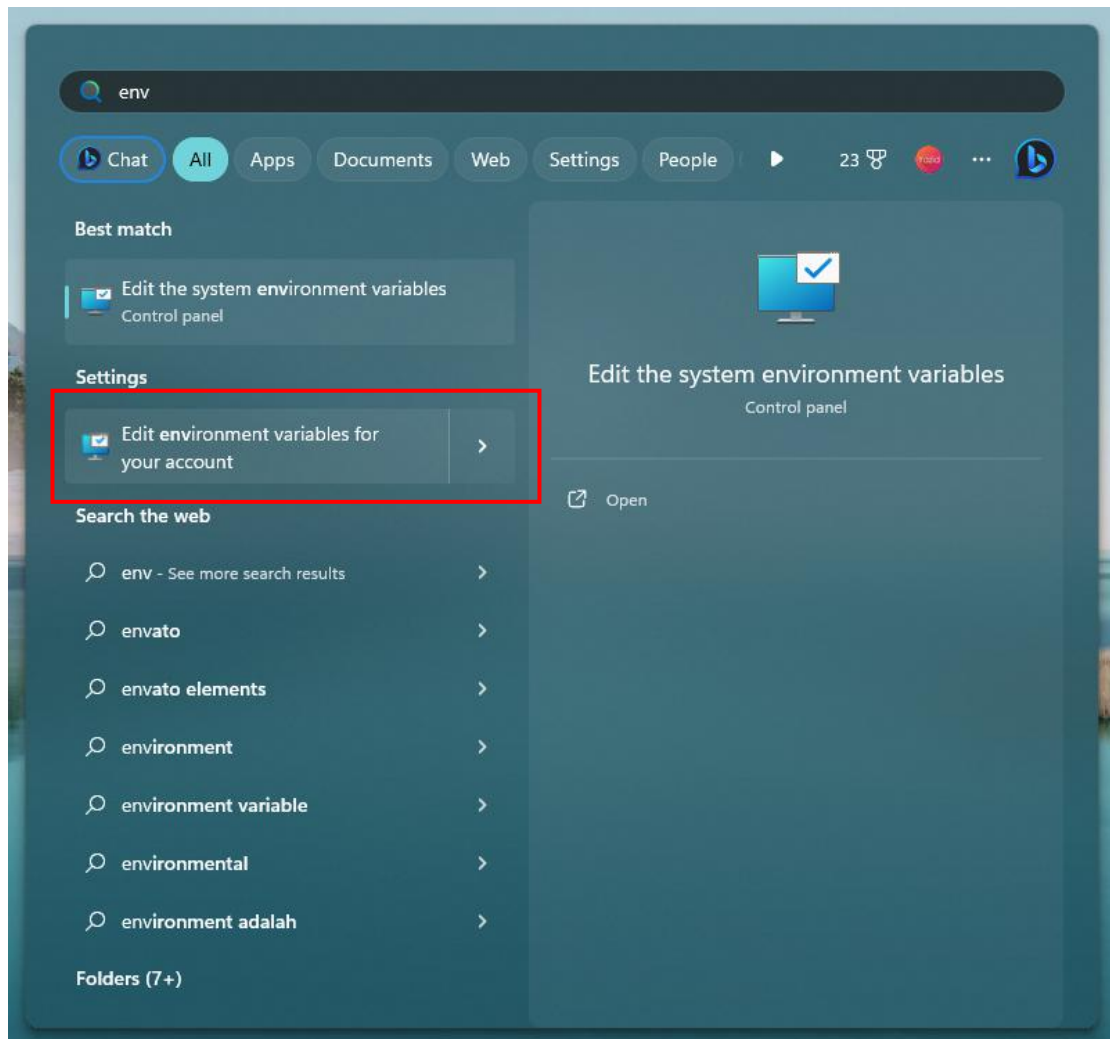
Gambar 21. Konfirmasi pengunduhan MUSCLE5 pada browser Microsoft Edge di Windows 11 bag.2

Setelah muscle berhasil diunduh, pindahkan file tersebut melalui file explorer ke dalam direktori yang anda inginkan, contohnya file muscle.exe dipindahkan pada direktori Muscle pada folder home user.



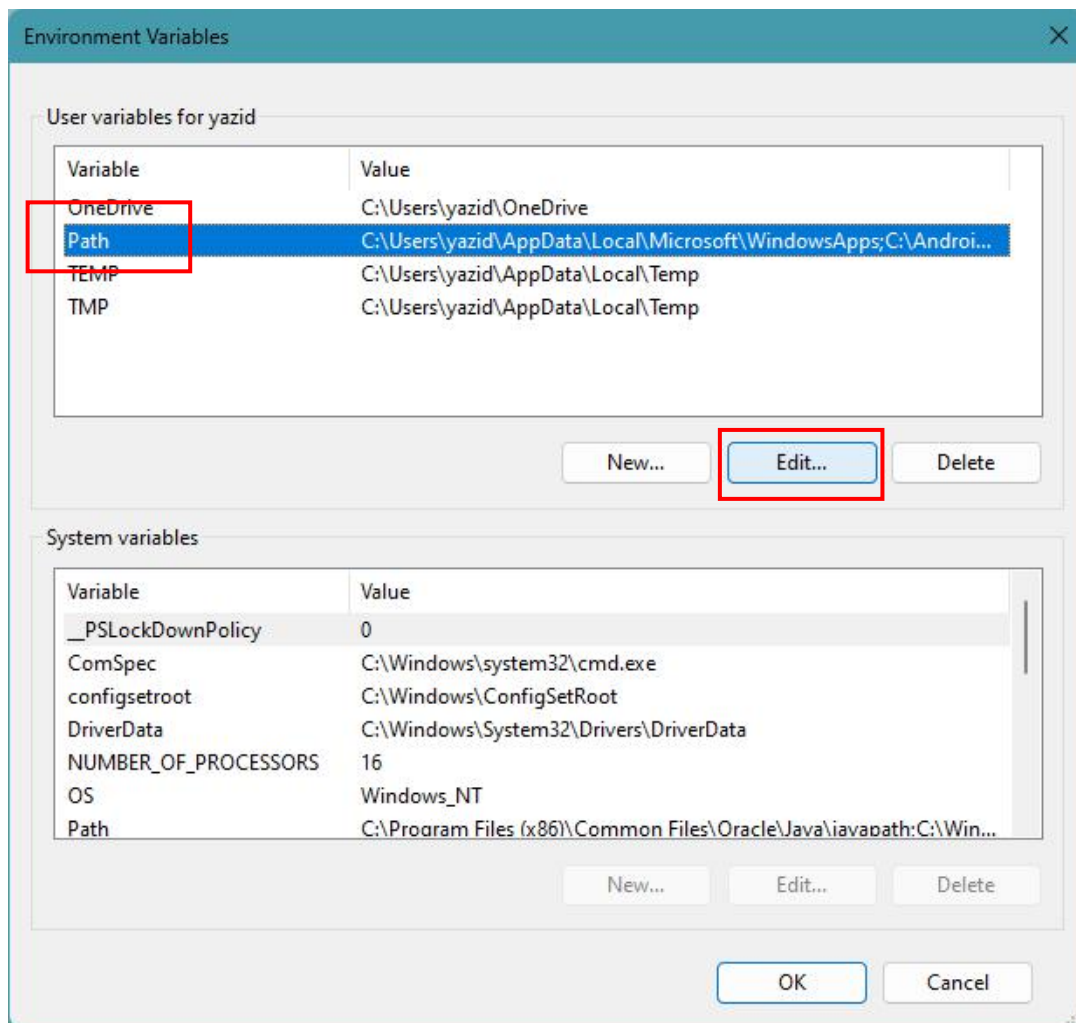
Gambar 22. Direktori instalasi muscle pada Windows

Kemudian, pengguna perlu mengkonfigurasi terlebih dahulu environment variables menuju binary muscle dengan pertama membuka start menu dan mencari menggunakan kata kunci “env”. Pada opsi kedua pilih “Edit environment variables for your account” seperti yang ditunjukkan gambar 23.



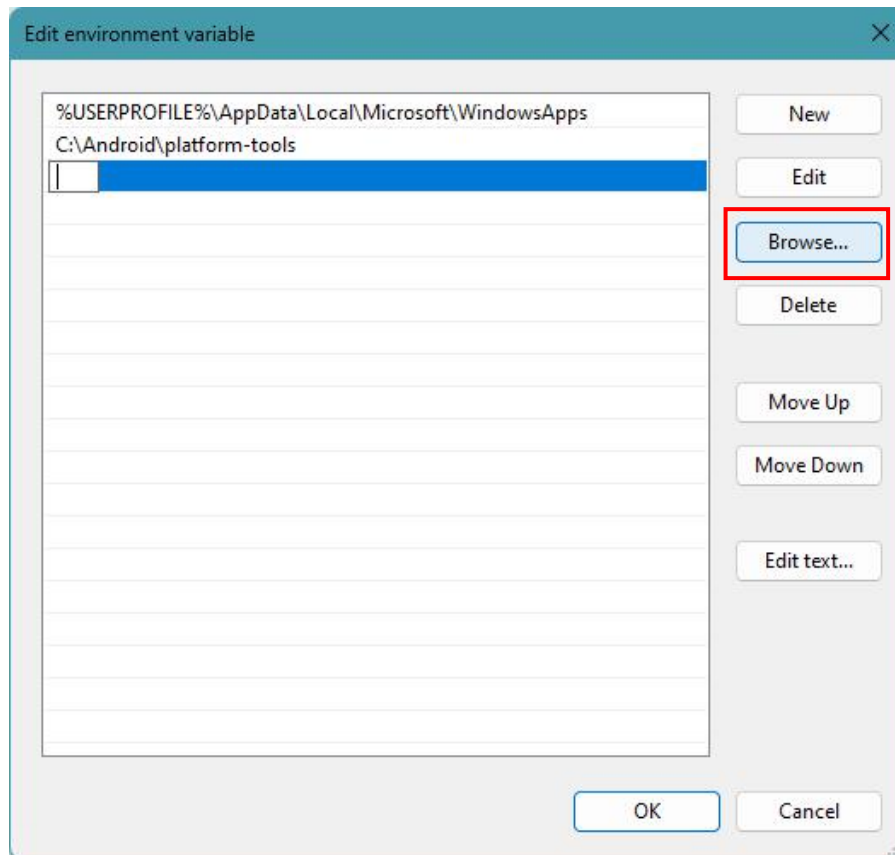
Gambar 23. Start menu windows untuk mengedit environment variables

Jendela environment variables kemudian akan terbuka, pengguna perlu mengubah Variable bernama Path seperti yang ditunjukkan gambar 24.

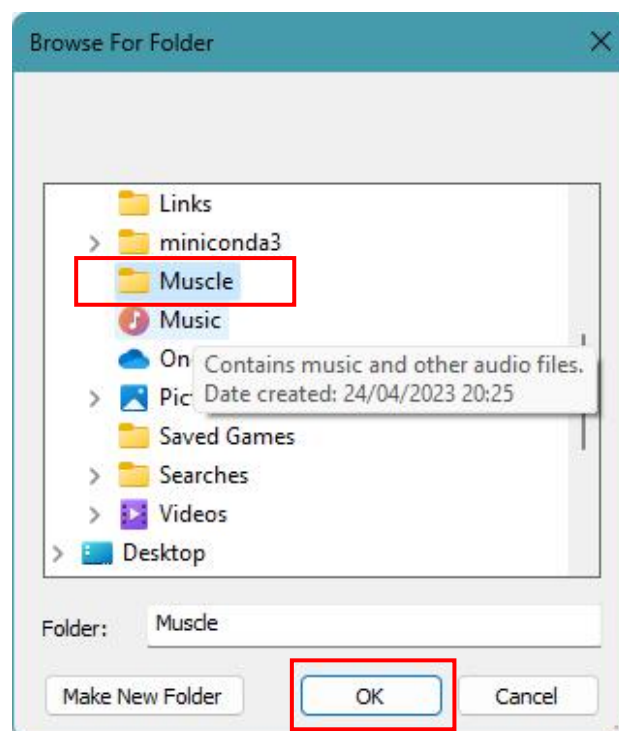


Gambar 24. Jendela konfigurasi environment variable

Ketika variable “Path” diedit, maka akan muncul daftar path yang telah dikonfigurasi sebelumnya. Klik tombol browse untuk menambahkan direktori baru dengan menavigasikan terlebih dahulu ke direktori instalasi Muscle seperti yang ditunjukkan pada gambar 25 dan 26.

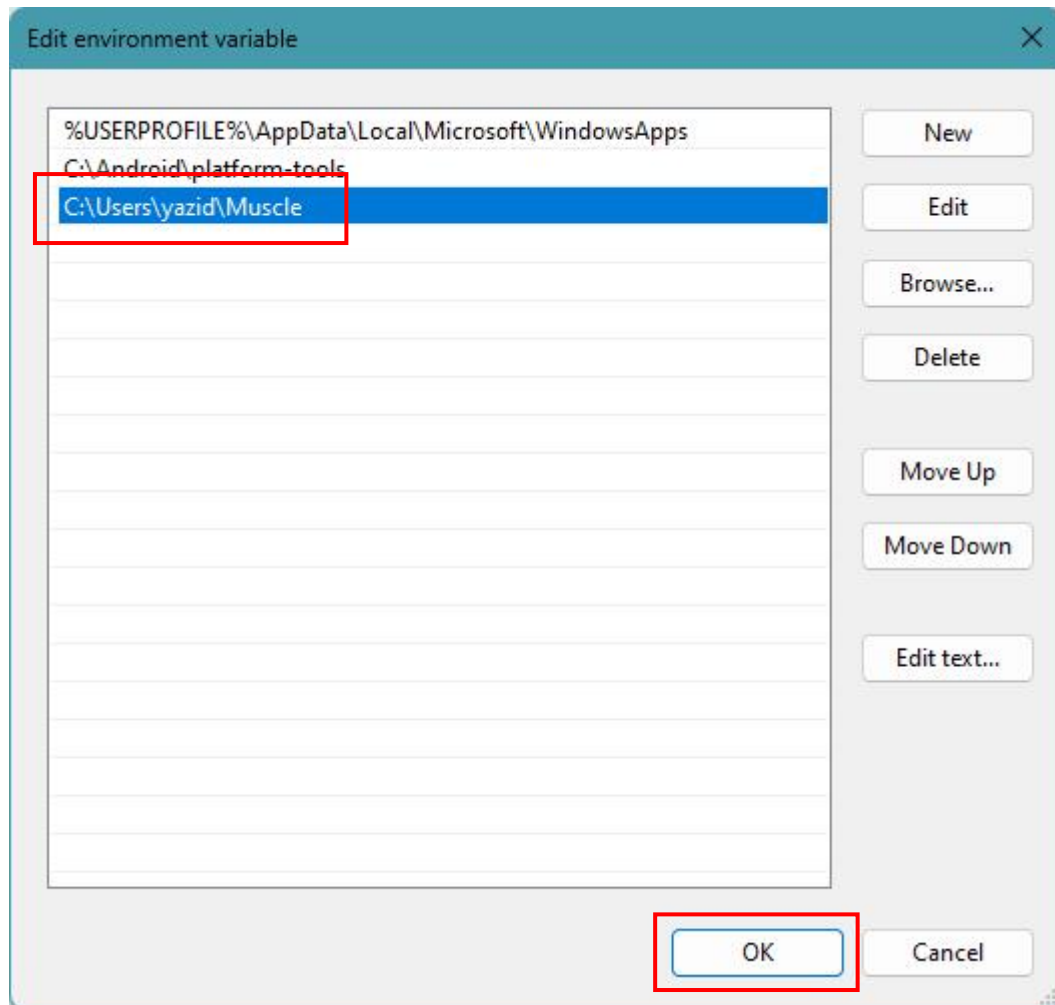


Gambar 25. Jendela untuk mengubah konfigurasi variable “Path”

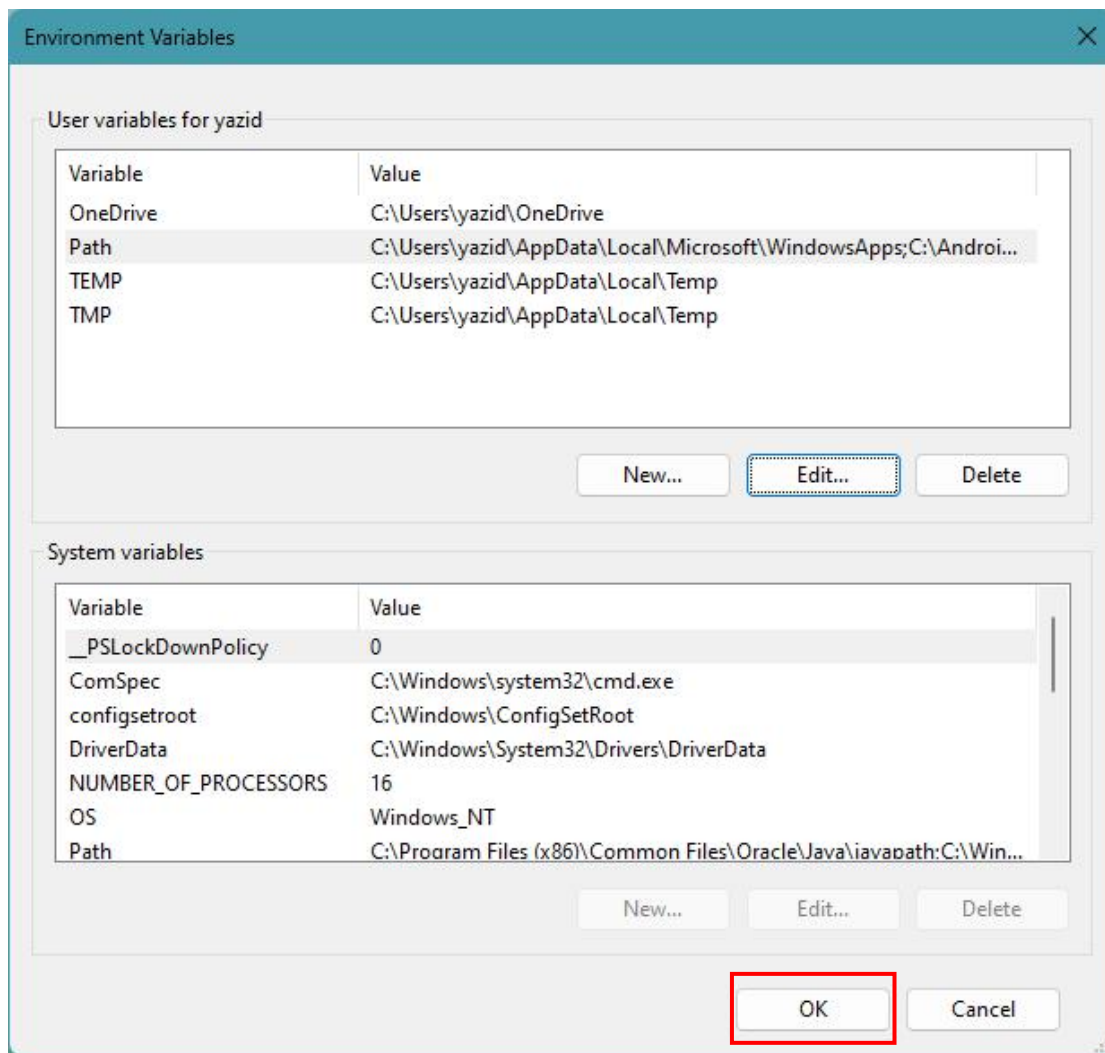


Gambar 26. Jendela pemilihan direktori baru pada variable “Path” menuju folder Muscle

Setelah direktori dipilih maka akan muncul satu item baru pada variable "Path" yaitu direktori instalasi Muscle.

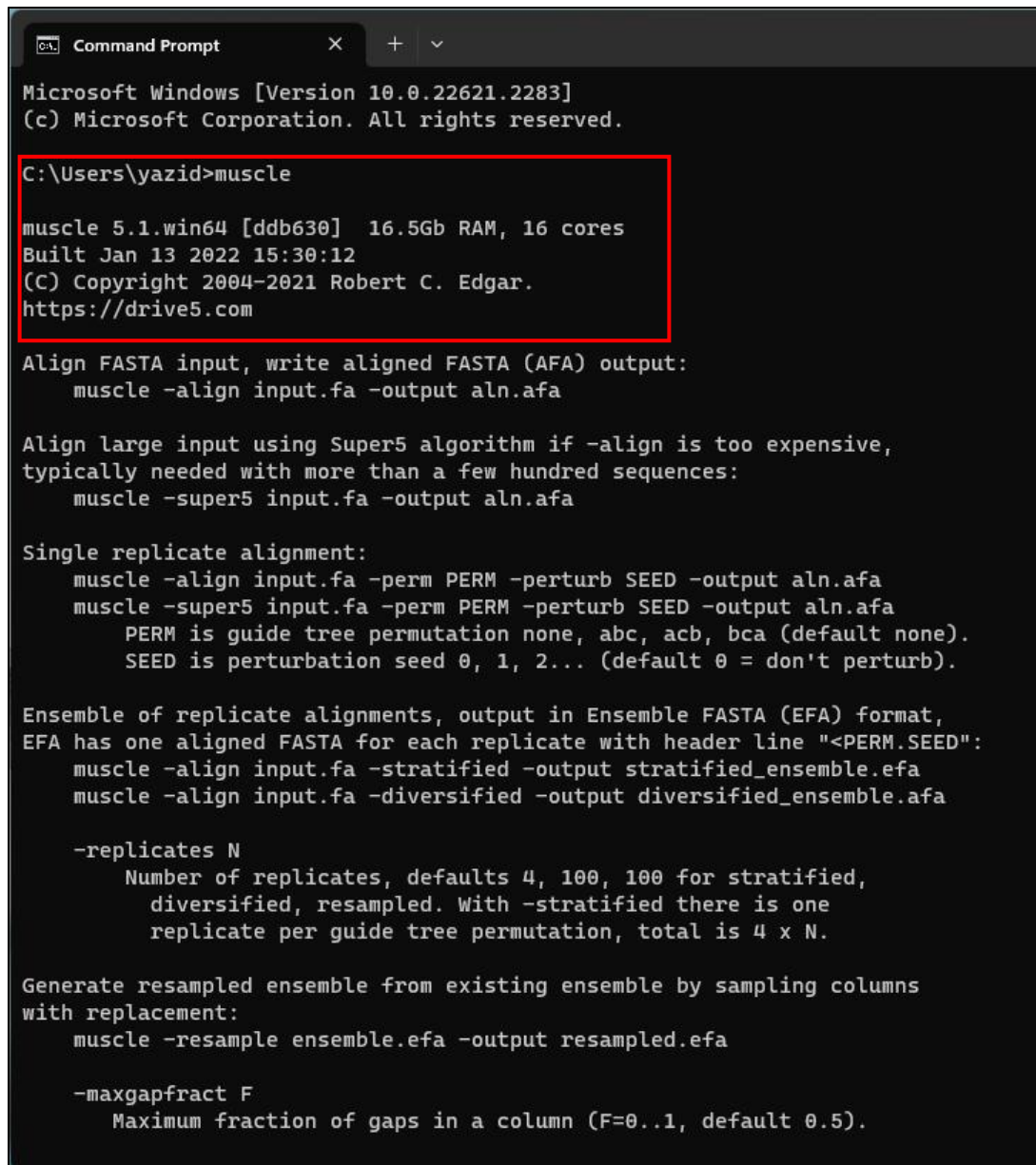


Gambar 27. Jendela variable "Path" setelah direktori Muscle ditambahkan



Gambar 28. Jendela konfigurasi environment variable setelah direktori Muscle ditambahkan pada variable Path

Pengguna cukup mengklik ok dari kedua window sesuai dengan gambar 27 dan 28.



```
Microsoft Windows [Version 10.0.22621.2283]
(c) Microsoft Corporation. All rights reserved.

C:\Users\yazid>muscle

muscle 5.1.win64 [ddb630] 16.5Gb RAM, 16 cores
Built Jan 13 2022 15:30:12
(C) Copyright 2004-2021 Robert C. Edgar.
https://drive5.com

Align FASTA input, write aligned FASTA (AFA) output:
    muscle -align input.fa -output aln.afa

Align large input using Super5 algorithm if -align is too expensive,
typically needed with more than a few hundred sequences:
    muscle -super5 input.fa -output aln.afa

Single replicate alignment:
    muscle -align input.fa -perm PERM -perturb SEED -output aln.afa
    muscle -super5 input.fa -perm PERM -perturb SEED -output aln.afa
    PERM is guide tree permutation none, abc, acb, bca (default none).
    SEED is perturbation seed 0, 1, 2... (default 0 = don't perturb).

Ensemble of replicate alignments, output in Ensemble FASTA (EFA) format,
EFA has one aligned FASTA for each replicate with header line "<PERM.SEED":
    muscle -align input.fa -stratified -output stratified_ensemble.efa
    muscle -align input.fa -diversified -output diversified_ensemble.afa

    -replicates N
        Number of replicates, defaults 4, 100, 100 for stratified,
        diversified, resampled. With -stratified there is one
        replicate per guide tree permutation, total is 4 x N.

Generate resampled ensemble from existing ensemble by sampling columns
with replacement:
    muscle -resample ensemble.efa -output resampled.efa

    -maxgapfract F
        Maximum fraction of gaps in a column (F=0..1, default 0.5).
```

Gambar 29. Program Muscle dapat diakses pada command prompt Windows setelah konfigurasi environment variable selesai

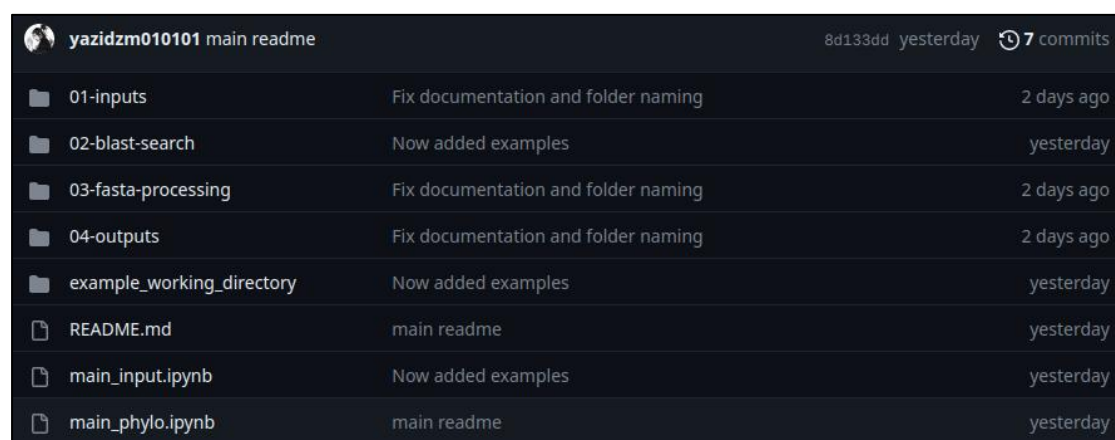
Dengan langkah ini dilakukan seharusnya program Muscle dapat dipanggil melalui command prompt atau powershell prompt.











BAB II

REPOSITORI PROYEK, FASTA DATASET, DAN NCBI

H. Repositori Proyek

Repositori proyek, dataset, dan python notebook yang digunakan pada penelitian dapat diakses melalui tautan https://github.com/yazidzm010101/sars_cov2-blast-recursive-phylogenetic.



 yazidzm010101 main readme	8d133dd yesterday	 7 commits
 01-inputs	Fix documentation and folder naming	2 days ago
 02-blast-search	Now added examples	yesterday
 03-fasta-processing	Fix documentation and folder naming	2 days ago
 04-outputs	Fix documentation and folder naming	2 days ago
 example_working_directory	Now added examples	yesterday
 README.md	main readme	yesterday
 main_input.ipynb	Now added examples	yesterday
 main_phylo.ipynb	main readme	yesterday

Gambar 30. Repositori proyek penelitian

Pada folder repositori terdapat dua python notebook yang dapat digunakan yaitu `main_input.ipynb` dan `main_phylo.ipynb`. Folder yang diawali dengan angka adalah folder yang sengaja disimpan sebagai aset dari penelitian utama. Sedangkan folder bernama `example_working_directory` digunakan sebagai folder yang dijadikan sebagai contoh direktori kerja yang digunakan pada python notebook yang telah disediakan.

I. Sars-Cov-2 Wuhan FASTA

Pada proyek ini, file utama untaian virus SARS-CoV-2 yang dijadikan sebagai data utama percobaan didapat dari situs NCBI (<https://www.ncbi.nlm.nih.gov/sars-cov-2/>) beserta dengan informasi Gene ID (https://www.ncbi.nlm.nih.gov/gene/?term=NC_045512) seperti yang ditunjukkan pada gambar di bawah ini.

Tabular ▾ 20 per page ▾ Sort by Relevance ▾ Send to: ▾

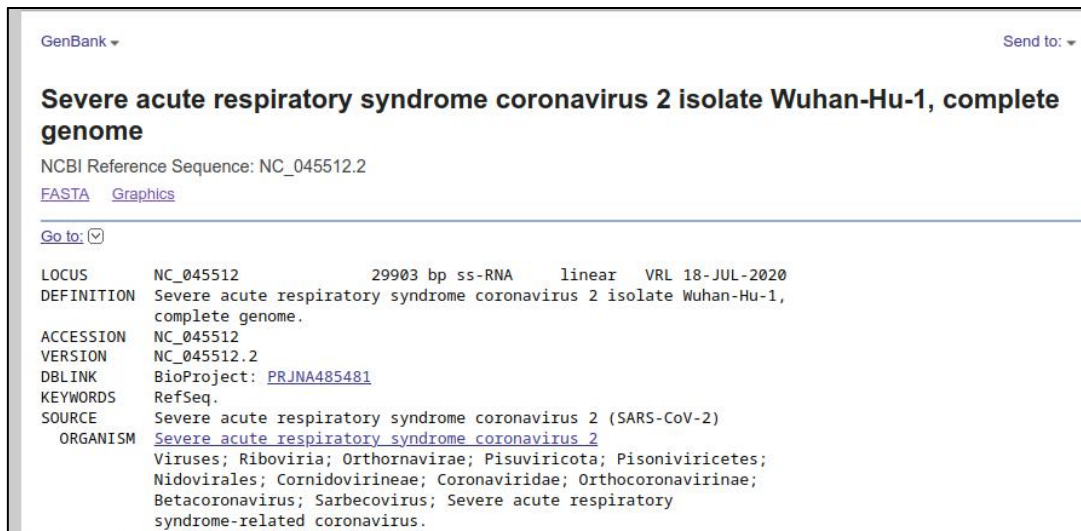
[Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome](#)
29,903 bp genomic DNA.
Isolate: Wuhan-Hu-1. Host: Homo sapiens. Old_name: Wuhan seafood market pneumonia virus. Gb_acronym: SARS-CoV-2. Country: China. Collection_date: Dec-2019.
Accession: NC_045512.2 GI: 1798174254
[GenBank](#) [FASTA](#) [Graphics](#)

Search results
Items: 11
[See also 10 discontinued or replaced items.](#)

Name/Gene ID	Description	Location	Aliases
<input type="checkbox"/> ORF1ab ID: 43740578	ORF1a polyprotein; ORF1ab polyprotein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (266..21555)	GU280_gp01
<input type="checkbox"/> S ID: 43740568	surface glycoprotein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (21563..25384)	GU280_gp02, spike glycoprotein
<input type="checkbox"/> N ID: 43740575	nucleocapsid phosphoprotein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (28274..29533)	GU280_gp10
<input type="checkbox"/> ORF8 ID: 43740577	ORF8 protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (27894..28259)	GU280_gp09
<input type="checkbox"/> E ID: 43740570	envelope protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (26245..26472)	GU280_gp04
<input type="checkbox"/> ORF3a ID: 43740569	ORF3a protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (25393..26220)	GU280_gp03
<input type="checkbox"/> M ID: 43740571	membrane glycoprotein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (26523..27191)	GU280_gp05
<input type="checkbox"/> ORF7a ID: 43740573	ORF7a protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (27394..27759)	GU280_gp07
<input type="checkbox"/> ORF6 ID: 43740572	ORF6 protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (27202..27387)	GU280_gp06
<input type="checkbox"/> ORF10 ID: 43740576	ORF10 protein [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (29558..29674)	GU280_gp11
<input type="checkbox"/> ORF7b ID: 43740574	ORF7b [Severe acute respiratory syndrome coronavirus 2]	NC_045512.2 (27756..27887)	GU280_gp08

Gambar 31. Halaman Gene ID dari SARS-CoV-2 NC_045512

Pada halaman tersebut dapat dimuat informasi detail untaian genetiknya dengan mengklik judul yang diberi dengan tanda kotak merah sehingga diarahkan pada halaman yang ditunjukkan pada gambar di bawah ini. (<https://www.ncbi.nlm.nih.gov/nuccore/1798174254/>).



Gambar 32. Halaman detail informasi genetika dari virus SARS-CoV-2

Halaman ini memuat lengkap informasi serta metadata genetika dari virus SARS-CoV-2, pada gambar diatas (A) locus menandakan ID rangkaian yang bersifat unik pada GenBank NCBI, dan (B) menandakan ukuran bp (base pair/pasangan basa). Antarmuka dari halaman ini juga menyediakan akses cepat untuk membuka halaman BLAST query dengan menggunakan ID dari sekuens yang sedang dimuat sebagai parameter pencarian.



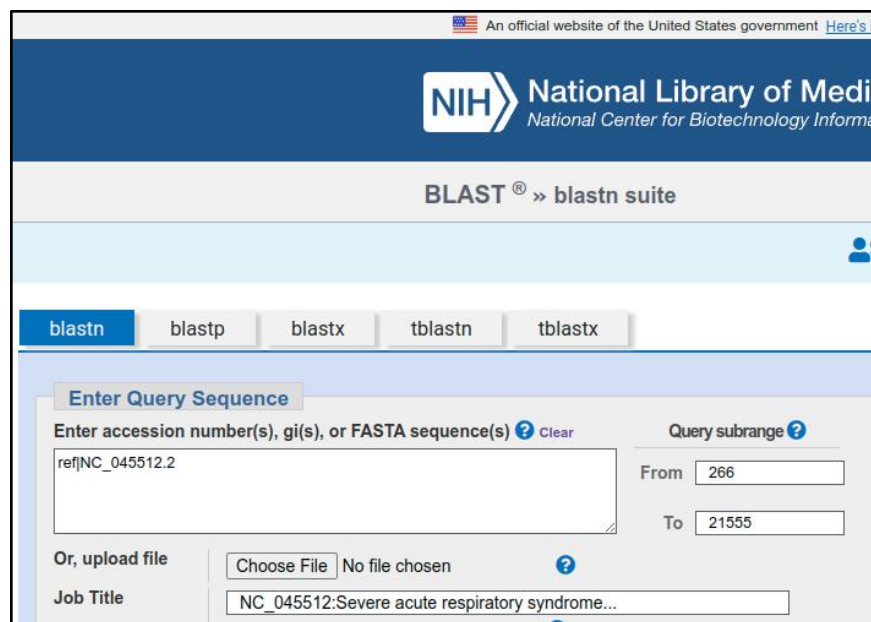
Gambar 33. Navigasi pintas untuk menjalankan BLAST menggunakan spesifik halaman sekuens yang sedang dibuka

BAB III

IMPLEMENTASI

J. BLAST query

Kode sekuens yang digunakan pada poin F kemudian dapat dijadikan sebagai parameter inquiry pada program blastn yang diakses pada halaman berikut.



The screenshot shows the NCBI BLAST suite interface. At the top, there is a header for the National Library of Medicine (NIH) and the National Center for Biotechnology Information. Below this, the BLAST logo and 'blastn suite' are displayed. The 'blastn' tab is selected among other options like 'blastp', 'blastx', 'tblastn', and 'tblastx'. The main form is titled 'Enter Query Sequence'. It has a text input field for 'Enter accession number(s), gi(s), or FASTA sequence(s)' with the value 'ref|NC_045512.2'. To the right of this field is a 'Clear' button. Below the input field is a section for 'Or, upload file' with a 'Choose File' button and the text 'No file chosen'. To the right of the input field is a 'Query subrange' section with 'From' and 'To' fields, containing the values '266' and '21555' respectively. At the bottom, there is a 'Job Title' field with the text 'NC_045512:Severe acute respiratory syndrome...'.

Gambar 34. Halaman BLAST diarahkan dari detail genetik SARS-CoV-2

Sebagai sampel kode orf1ab pada fasta wuhan terletak antara 266 dan 21555. Pada area atau subarea tersebut dapat dijadikan sebagai parameter query subrange untuk mencari sekuens yang similar.

K. Mengunduh hasil pencarian BLAST

Setelah proses query selesai dijalankan, antarmuka web NCBI akan menampilkan halaman daftar hasil pencarian berupa tabel. Untuk mengunduh hasil pencarian ke dalam single fasta dapat menavigasi terlebih dahulu hasil pencarian ke dalam format genbank.

of the United States government [Here's how you know](#)

National Library of Medicine

U.S. Department of Health and Human Services
National Center for Biotechnology Information

Nucleotide

Advanced

Summary ▾ 20 per page ▾ Sort by Default order ▾

Items: 1 to 20 of 100

<< First < Prev P

☐ [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
 1. 29,882 bp linear RNA
 Accession: OX877548.1 GI: 2522355244
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

☐ [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
 2. 29,850 bp linear RNA
 Accession: OX877547.1 GI: 2522355243
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

☐ [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
 3. 29,835 bp linear RNA
 Accession: OX877546.1 GI: 2522355242
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

☐ [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
 4. 29,903 bp linear RNA
 Accession: OX877545.1 GI: 2522355241
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

☐ [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
 5. 29,850 bp linear RNA
 Accession: OX877544.1 GI: 2522355240
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

☒ Complete Record
☐ Coding Sequences
☐ Gene Features
Choose Destination
☒ File ☐ Clipboard
☐ Collections
 Download 100 items.
 Format
 FASTA ▾
 Sort by
 Default order ▾
 Show GI ☐

Severe acute respiratory syndrome coronavirus 2

Gambar 35. Halaman BLAST diarahkan dari detail genetik SARS-CoV-2

NCBI menyediakan format penyimpanan sebagai complete record ataupun dipotong berdasarkan coding sequence. Adapun dalam penelitian ini metadata informasi coding sequence sangat penting untuk mengekstraksi sekuens berdasarkan letak indeks protein. Seperti yang ditunjukkan pada gambar 36, kita dapat

memilih opsi coding sequence untuk diunduh menjadi satu file fasta yang telah dipecah struktur coding sequencenya.

The screenshot shows the NCBI GenBank website interface. At the top, there is a header for the National Library of Medicine and the Center for Biotechnology Information. Below the header, there is a search bar and a dropdown menu for 'Nucleotide'. A list of items is displayed, showing details for several SARS-CoV-2 genome assemblies. A dropdown menu is open, showing options to download 'Complete Record', 'Coding Sequences' (selected), or 'Gene Features'. The 'Coding Sequences' option is selected, and the 'Format' is set to 'FASTA Nucleotide'. The 'Create File' button is visible.

Items: 1 to 20 of 100

1. [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
29,882 bp linear RNA
Accession: OX877548.1 GI: 2522355244
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

2. [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
29,850 bp linear RNA
Accession: OX877547.1 GI: 2522355243
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

3. [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
29,835 bp linear RNA
Accession: OX877546.1 GI: 2522355242
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

4. [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
29,903 bp linear RNA
Accession: OX877545.1 GI: 2522355241
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

5. [Severe acute respiratory syndrome coronavirus 2 genome assembly, complete genome: monopartite](#)
29,850 bp linear RNA
Accession: OX877544.1 GI: 2522355240
[BioProject](#) [BioSample](#) [Taxonomy](#)
[GenBank](#) [FASTA](#) [Graphics](#)

Gambar 36. Halaman BLAST diarahkan dari detail genetik SARS-CoV-2

L. Menyatukan FASTA input dan FASTA Hasil Pencarian

main_input.ipynb	
<pre>seq_search = read_fasta('./02-blast- search/search_result_1-cds.fasta')</pre>	
<pre>seq_input = read_fasta('./01-inputs/NC_045512.2- cds.fasta')</pre>	
<pre>seq_merged = merge_sequences_list([seq_search, seq_input])</pre>	

Perintah di atas memanggil dua fungsi yang telah didefinisikan pada python notebook pada “main_input.ipynb” yaitu “read_fasta” dan “merge_sequence_list”. Dimana “read_fasta” menerima satu parameter/argumen berupa filepath menuju fasta yang ingin dibaca menjadi daftar sequence records ke dalam variable “seq_search” dan “seq_input”. Filepath yang dijadikan parameter juga dapat dibuat menjadi relatif terhadap direktori yang sedang aktif dengan menambahkan karakter “.” di depan. Perlu diingat sintaks diatas ditulis pada sistem operasi linux dimana setiap subfolder ditandai dengan simbol garis miring “/” sedangkan sistem operasi windows menggunakan simbol garis miring terbalik “\”, walaupun pada kasus dokumentasi ini subfolder dengan garis miring “/” dapat berjalan dengan baik pada Windows 11.

M. Ekstraksi Coding Sequence dari FASTA menggunakan biopython

Masih berlanjut dengan langkah sintaks sebelumnya, sekuens yang telah di-combine kemudian dapat diekstrak berdasarkan nama proteinnya dengan memanggil fungsi “filter_sequences_by_protein”. Fungsi ini dirancang khusus untuk membaca sekuens yang di-parse dari FASTA sekuens dalam format CDS (Coding Sekuens) karena memiliki informasi/atribut mengenai gene/protein pada definisi FASTA header. Dengan memanfaatkan

pustaka regex pada python fungsi ini menerima dua parameter yaitu list of SeqRecord dan kedua adalah nama proteinnya.

main_input.ipynb

```
seq_orf1ab = filter_sequences_by_protein(seq_merged,  
"ORF1ab polyprotein")
```

N. Pemangkasan file FASTA menggunakan BioPython

Satu sekuens dari ORF1ab sendiri sudah cukup panjang, dan pada studi kasus ini ketika dicoba selaraskan dengan program MUSCLE sekalipun tidak menghasilkan keluaran apapun karena prosesnya cukup berat sampai terminal/command prompt memaksa berhenti proses tersebut dengan status exit sigterm (pada sistem operasi linux dengan desktop environment GNOME). Bahkan pada aplikasi terminal lain (konsole pada desktop environment KDE Plasma) status ini tidak muncul dan jendela terminal langsung terpaksa tertutup dengans sendirinya. Pada python notebook ini terdapat fungsi bernama "cut_sequences_length" untuk memangkas setiap seqRecord pada list dari indeks ke nol sampai maksimum angka yang dijadikan paremeter pada pemanggilan fungsi.

main_input.ipynb

```
seq_orf1ab_1000 = cut_sequences_length(seq_orf1ab,  
1000)
```

Menggunakan lambda function dari fungsi map dapat dipotong menggunakan indeks 0 sampai 1000 sehingga hanya 1000 asam basa pertama yang disimpan pada setiap sekuens.

O. Penyelarasan FASTA menggunakan MUSCLE

List SeqRecord yang telah dipangkas kemudian dapat disimpan ke dalam direktori dengan memanggil perintah `write_sequence_into_fasta` yang menerima dua parameter, parameter pertama adalah objek list SeqRecord, parameter kedua adalah filename path yang akan digunakan sebagai penyimpanan file teks fasta.

main_phylo.ipynb

```
write_sequence_into_fasta(seq_orf1ab_1000,  
    "./example_working_directory/search_result_1_orf1ab_1000.fasta")
```

File yang telah dipangkas sebelumnya kemudian dapat diselaraskan menjadi file fasta juga menggunakan program MUSCLE dengan perintah berikut melalui command prompt/terminal.

terminal/command prompt

```
muscle -align input.fasta -output output.fasta
```

Atau pada python notebook juga telah disediakan satu fungsi serupa “`align_using_muscle`” yang menerima parameter filepath yang ingin diselaraskan. Secara default nama file output akan memiliki akhiran “`_muscle.fasta`” pada direktori yang sama. Perlu diingat fungsi ini hanya dapat berjalan apabila langkah pada poin G telah dilakukan dengan baik.

main_phylo.ipynb

```
align_using_muscle("./example_working_directory/search_result_1_orf1ab_1000.fasta")
```

P. Pembentukan pohon filogenetik

Pohon filogenetik kemudian dapat dibentuk dengan pertama membaca file fasta yang telah diselaraskan untuk dihitung terlebih dahulu distance matrixnya.

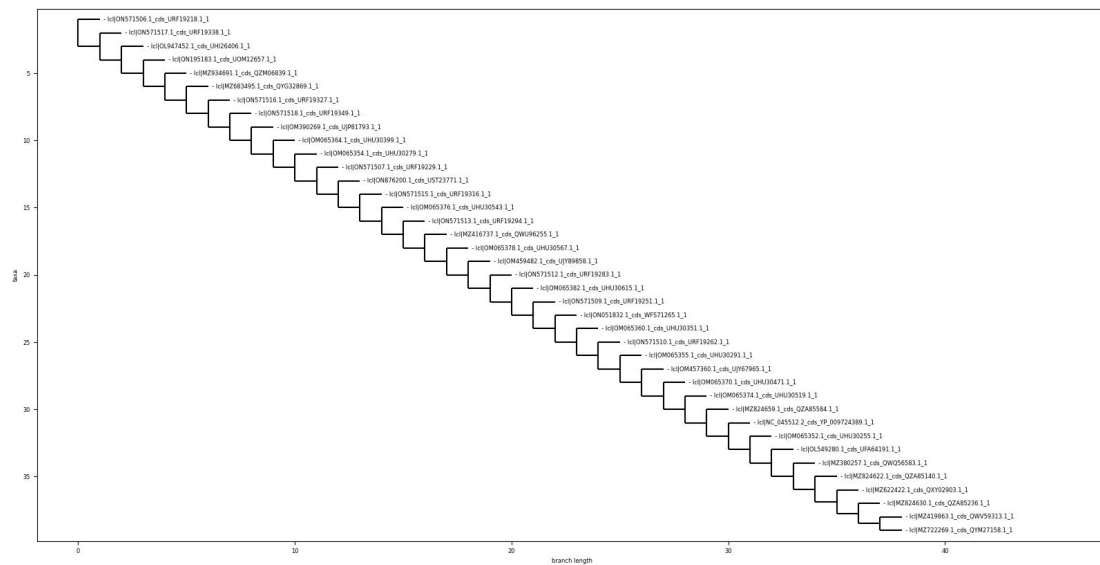
main_phylo.ipynb

```
aln =  
read_alignment("./example_working_directory/search_resu  
lt_1_orf1ab_1000_muscle.fasta")  
  
dis = calculate_distance_matrix(aln)  
tree = construct_tree(dis, "upgma")  
tree.ladderize()
```

Pohon yang telah dikonstruksi kemudian dapat ditampilkan visualisasinya dengan bantuan matplotlib sesuai dengan sintaks di bawah ini.

main_phylo.ipynb

```
matplotlib.rc('font', size=6)  
fig = plt.figure(figsize=(20, 10), dpi=100)  
axes = fig.add_subplot(1, 1, 1)  
  
Phylo.draw(tree, axes=axes,  
label_func=label_func_non_zero_alt, do_show=False)
```

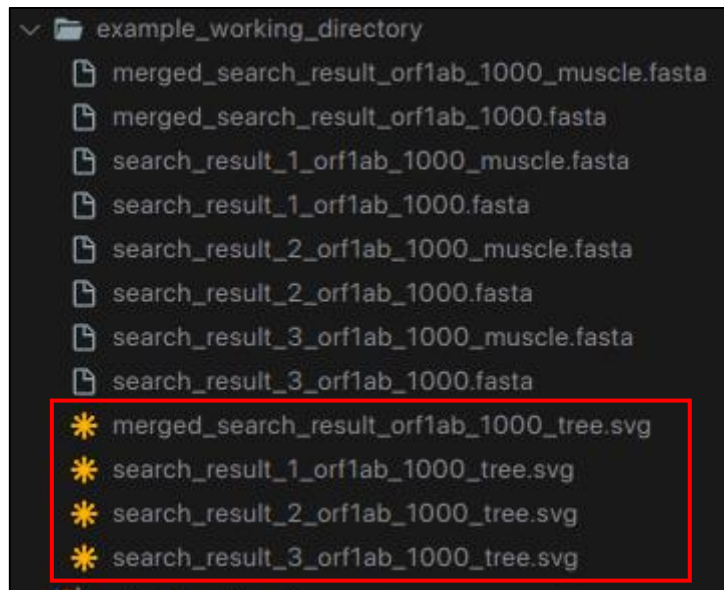


Gambar 37. Contoh pohon filogenetik pada search result 1

Pohon yang telah dibentuk kemudian dapat ditampilkan sebagai gambar statis dan juga dapat disimpan ke dalam media penyimpanan dalam format svg vector dengan perintah di bawah ini.

main_phylo.ipynb

```
pylab.savefig('./example_working_directory/search_result_1_orflab_1000_tree.svg',format='svg',
bbox_inches='tight', dpi=300)
```



Gambar 38. Pohon filogenetik juga dapat disimpan ke dalam file
svg

Q. Melabeli pohon filogenetik dengan Complete FASTA file

Pohon filogenetik yang dibangun juga dapat menerima label function untuk menentukan format penamaan setiap simpul. Dalam kasus ini, sudah disediakan satu fungsi dalam python notebook untuk membaca file complete FASTA sebagai referensi utama penamaan simpul setiap pohon.

```
main_phylo.ipynb

aln =
read_alignment("./example_working_directory/search_resu
lt_1_orf1ab_1000_muscle.fasta")

dis = calculate_distance_matrix(aln)

tree = construct_tree(dis, "upgma")

tree.ladderize()
```

```

matplotlib.rc('font', size=6)

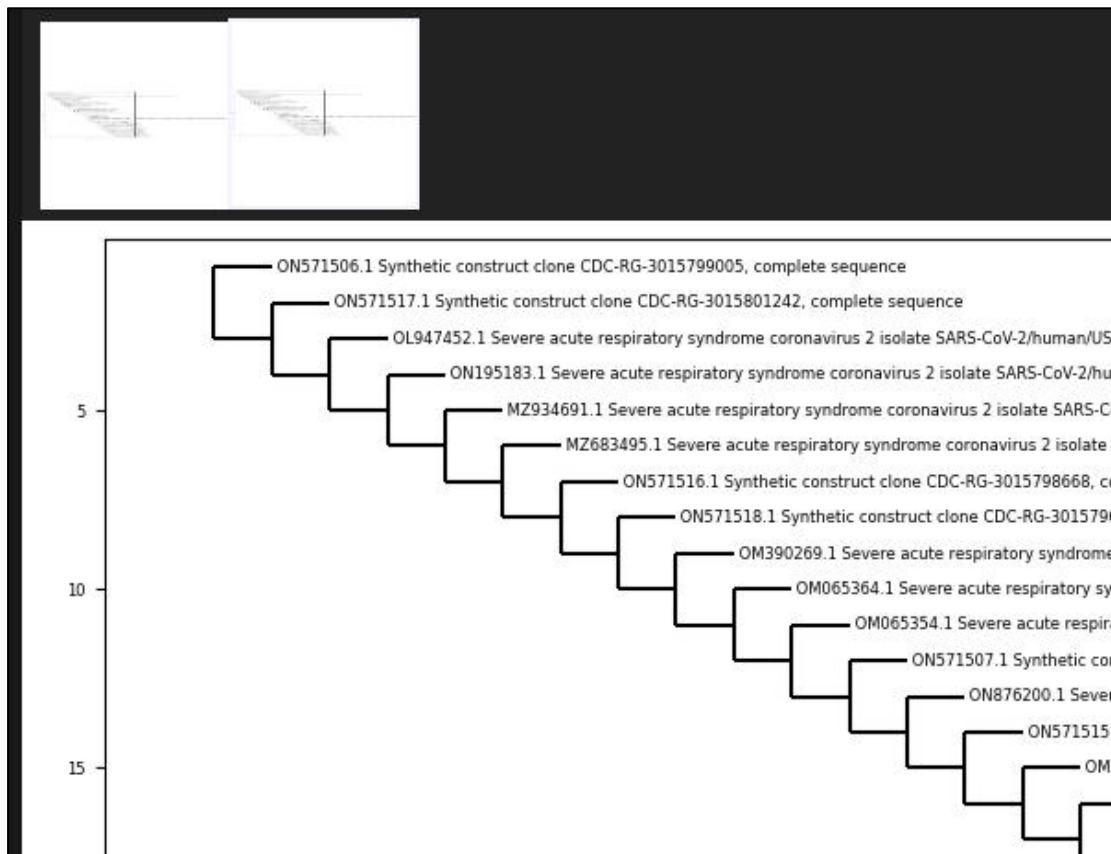
fig = plt.figure(figsize=(20, 10), dpi=100)

axes = fig.add_subplot(1, 1, 1)

ref = generate_sequences_headers_ref_labels([
    SeqIO.parse("./02-blast-
search/search_result_1.fasta", "fasta")
])

Phylo.draw(tree, axes=axes, label_func=lambda c:
label_func_by_seqids(c, ref), do_show=False)

```



Gambar 39. Pohon filogenetik juga dapat ditampilkan penamaan simpul menggunakan custom function

R. Pencarian Rekursif

Pencarian rekursif kemudian dapat dilakukan dengan cara memilih salah satu di antara sekuens yang digunakan pada langkah sebelumnya untuk dijadikan sebagai parameter pencarian BLAST pada iterasi berikutnya. Kemudian langkah yang sama dari poin J sampai poin Q dapat dilakukan kembali.

S. Penggabungan pohon filogenetik

Pohon filogenetik yang telah terbentuk pada setiap percobaan dapat digabungkan kembali untuk melihat gambaran yang lebih besar dengan cara menyatukan setiap fasta input dan fasta hasil pencarian ke dalam satu variable list untuk kemudian di-filter berdasarkan daftar pengecualian SeqID menggunakan fungsi “filter_sequences_list_by_ids”. Fungsi ini menerima dua parameter, parameter pertama adalah array of seqRecord list dan parameter kedua adalah daftar SeqId yang digunakan pada penggabungan sekuens. Dan menerima nilai balik berupa list of SeqRecord dan dapat diproses kembali dengan tahapan yang sama melalui poin M sampai Q.

main_input.ipynb
<pre>seq_list = []</pre>
<pre>seq_list.append(read_fasta('./02-blast- search/search_result_1-cds.fasta')) seq_list.append(read_fasta('./01-inputs/NC_045512.2- cds.fasta'))</pre>
<pre>seq_list.append(read_fasta('./02-blast- search/search_result_2-cds.fasta')) seq_list.append(read_fasta('./01-inputs/OM065360.1- cds.fasta'))</pre>

```
seq_list.append(read_fasta('./02-blast-  
search/search_result_3-cds.fasta'))
```

```
seq_list.append(read_fasta('./01-  
inputs/OR184938.1_cds.fasta'))
```

```
exclusion_ids = [  
    'OR182755.1',  
    'OR182751.1',  
    'OR182753.1',  
    'OR182737.1',  
    'OR182738.1',  
    'OR184934.1',  
    'OR182735.1',  
    'OR184938.1',  
    'OR183340.1',  
    'OR183341.1',  
    'OM065360.1',  
    'OR184936.1',  
    'OR184932.1',  
    'NC_045512.2',  
    'MZ722269.1',  
    'OM065364.1'
```

```
]
```

```
merged_seq = filter_sequences_list_by_ids(seq_list,  
exclusion_ids)
```

```
merged_seq_orf1ab =  
filter_sequences_by_protein(merged_seq, "ORF1ab  
polyprotein")  
  
merged_seq_orf1ab_1000 =  
cut_sequences_length(merged_seq_orf1ab, 1000)  
  
write_sequence_into_fasta(merged_seq_orf1ab_1000,  
"./example_working_directory/merged_search_result_orf1a  
b_1000.fasta")
```