|  |  |
| --- | --- |
| **Assignment Case** | Diagram  Description automatically generated |
| SCIE6062001 | SCIE6062016 | SCIE6062049  Computational Biology |
| **Computer Science** | **E233-SCIE6062-GU01-01** |
| ***Valid on*** *Even Semester Year 2022/2023* | **Revision 01** |

1. Mahasiswa tidak diperkenankan untuk:

*Students are prohibited from:*

* + - Berdiskusi dan/atau bekerja sama dengan mahasiswa lainnya,

*Discussing and/or cooperating with other students,*

* + - Melihat sebagian atau seluruh jawaban mahasiswa lainnya,

*Seeing a part or the whole answer from other students,*

* + - Membuka dan/atau menyalin jawaban dari buku, catatan, video, dan jenis referensilainnya,

*Open and/or copy answer from books, notes, videos, and other references,*

* + - Membuka dan/atau menyalin jawaban dari internet,

*Open and/or copy answer from the internet,*

* + - Mengumpulkan jawaban yang tidak sesuai dengan tema soal,

*Submitting an answer with a different theme from the given case,*

* + - Melakukan tindakan yang menyebabkan jawaban dicontek oleh orang lain atau kelompok lain, baik disengaja maupun tidak disengaja,

*Doing action that could result the answer being copied by someone or other groups, intentionally or unintentionally,*

* + - Melakukan tindakan kecurangan lainnya.

*Committing other dishonest actions.*

1. Jika mahasiswa dan/atau terbukti melakukan tindakan seperti yang dicantumkan pada butir ke-1, maka nilai mahasiswa dan/atau kelompok yang melakukan kecurangan, baik menyontek atau dicontek, akan dinolkan sesuai dengan peraturan yang berlaku.

*If it has been proven that a student and/or group has committed dishonest actions outlined in point 1 above, the students and/or groups related to the incident, regardless of which one copies or has their answer copied, will be issued a score of zero according to the regulation.*

1. Jawaban yang dapat diterima dan dinilai adalah jawaban yang dikumpulkan sebelum batas waktu yang telah ditentukan.

*The answer must be submitted before the designated deadline to be accepted and graded,*

1. Jawaban akan dinilai berdasarkan teknik yang diajarkan dalam praktikum dengan menggunakan software yang telah ditentukan.

*The scoring will be based on the materials taught during the practicum classes using the designated software. Using different software than requested may result in your answer not being graded.*

1. Jika Anda tidak membaca peraturan ini, maka Anda dianggap sudah membaca dan menyetujuinya.

*By taking this exam, you agree to these regulations, regardless of whether you have read it or not.*

1. Persentase penilaian untuk matakuliah ini adalah sebagai berikut:

*The score will be distributed as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 100% | - | - |

1. Perangkat lunak yang digunakan pada matakuliah ini adalah sebagai berikut:

*This course uses the following software:*

|  |
| --- |
| **Software**  *Software* |
| Anaconda 4.14.0  Jupyter Notebook  Python 3.7.6  Visual Studio Code |

1. Ekstensi file yang harus dikumpulkan untuk matakuliah ini adalah sebagai berikut:

*Your answers must be in the following file extensions:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| IPYNB | - | - |

## Soal

*Case*

For given **DNA sequences**:

**Sequence A** is obtained from **FASTA** **file** (**KY352407.fasta**).

**Sequence B** is obtained from **FASTA** **file** (**OQ297732.fasta**).

**Note**:

The **sequences** contain **ambiguous** **letters**. Use **seq = seq.replace(“<letter>”, “”)** function to remove the unwanted ambiguous **<letter>** from **seq**.

**Ambiguous** **letters**: [‘**N**’, ‘**Y**’, ‘**R**’, ‘**W**’, ‘**M**’, ‘**S**’, ‘**K**’]

Please use **Biopython** to do following operations:

1. **Sequence** **Manipulation**:
   1. **Find** and **display** the **total** **length** of **Sequence A** and **B**.
   2. **Find** and **display** which **index** the **first** **time** the codon (**CGC**) appears in **Sequence A** and **B**.
   3. **Take** the **first 6 nucleotides** from **Sequence A** and the **last** **9** **nucleotides** from **Sequence B** then **combine** the **sequences** and **reverse** it to a new sequence, **Sequence C**.
2. **Sequence** **Analysis** & **Plotting**:
3. **Determine** and **display** the **melting** **temperature** for **Sequence** **A**, **B**, and **C** (using **built-in** **Wallace** **Method**).
4. **Determine** and **display** the **GC** & **AT** **content** **percentage** for **Sequence** **A**, **B**, and **C**.
5. **Determine** and **display** the **molecular** **weight** for **Sequence** **A**, **B**, and **C**.
6. **Determine** and **display** the **nucleotides base** (**A**, **C**, **G**, **T**) **frequency** for **Sequence** **A**, **B**, and **C** with a **chart** using **matplotlib** library.
7. **DNA** & **mRNA** **Protein** **Synthesis**:
8. **Transcribe** and **display** the **DNA** **Sequence A**, **B**, and **C** into **mRNA Sequences**.
9. **Translate** and **display** the **mRNA** **Sequence A**, **B**, and **C** into **amino** **acids sequences**.
10. **Sequence Alignment & Similarities**:
11. **Perform** **local** **alignment** **pairwise** between **Sequence** **A** and Sequence **“TGTACATTATAATGCTCTATTTTCTCGC”** and **display** the **alignments**’ **score**.
12. **Perform** **local** **alignment** **pairwise** between **Sequence** **B** and Sequence **“CAGAGTAATGTGTACAGATATTGGGCTC”** and **display** the **alignments**’ **score**.
13. **Find** and **display** the **Levenshtein** **distance** between **Sequence** **A** and **B**.