National Technical University of Athens
School of Electrical and Computer Engineering
M.Sc. in Translational Engineering in Health and Medicine

Translational bioinformatics

bio1100

Report

EVANGELOS STAMOS

Supervisors: George Matsopoulos, Ioannis Makris
Professor NTUA | Teaching and Research Associate

Athens, February 2024

National Technical University of Athens
School of Electrical and Computer Engineering
M.Sc. in Translational Engineering in Health and Medicine

Translational bioinformatics

bio1100

Report

EVANGELOS STAMOS

Supervisors: George Matsopoulos, Ioannis Makris
Professor NTUA | Teaching and Research Associate

Approved by Prof. George Matsopoulos and Dr. Ioannis Makris in 14th February 2024.

(Signature)

.....
George Matsopoulos, Ioannis Makris
Professor NTUA | Teaching and Research Associate

Athens, February 2024

National Technical University of Athens
School of Electrical and Computer Engineering
M.Sc. in Translational Engineering in Health and Medicine

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this report, as well as the electronic files and source codes developed or modified in the course of this assingment, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this assignment or part of it does not belong to me because it is a product of plagiarism.

(Signature)

.....
Evangelos Stamos

14th February 2024

Abstract

In this assignment, we delve into the realm of translational bioinformatics with a focus on understanding genetic variations and their implications. Through practical exercises, we will employ bioinformatics tools to analyze genomes, identify mutations, and design primers for PCR amplification. Additionally, the assignment involves using BLAST for sequence alignment to elucidate genetic information relevant to health and disease. This project aims to equip us with the skills to bridge the gap between computational biology and clinical applications, highlighting the significance of bioinformatics in advancing personalized medicine.

Keywords

Bioinformatics, Translational, BLAST, Genomes, Mutations, Primers, PCR, Phylogenetic Trees

Table of Contents

List of Figures

List of Tables

Table 1. *Exercises implementation and short answers*

Question	Implementation	Answer(s)
1	LaTeX, Webcutter 2.0, Python	596 bp: BglII | 4 cuts: BstX2I, BstYI, MflI and XhoII
2	LaTeX, NCBI	Primers in 2.1
3	LaTeX, NCBI, UniProt	HBB: β thalassemia, Sequence 3.6, Structure 3.7, 3.8 PTM 3.9 Glutamic acid (E), lysine (K), leucine (L) and proline (P) 3.10
4	LaTeX, NCBI, MEGA11	Neighbor Joining Phylogenetic Tree 4.5, 4.11

Preface

This report was written during the Fall Semester of the academic year 2023 - 2024, in the context of project of the course bio1100 Translational bioinformatics.

In the following table all my personal information is included. For any question I am available on given email.

Category	Personal Data
Name	Evangelos Stamos
Registry Number	**03500050**
Student Type	Postgraduate student
Program Enrolled	**MSc Translational Engineering in Health and Medicine**
Email	stamosevangelos AT mail DOT ntua DOT gr

Report was implemented in LaTeX (pdfLaTeX) based on personal template specifically designed for course context. The table 1 gives detailed information about the implementation of each problem.

Chapter 1

Exercise 1

Given the following sequence:

CGAGGGACCTTTACGGCGTAATCCTGGAAACCATGACAAATCCAGAACCCCAAGGCTCCCCTCTTCA
GCTGATGTAGAATTTTGCCTGAGTTTGACCCATGGAAGGATTTGCTAGTCCACTTACTGGGATAGCGGATG
CCTCTCAAAGAGCATGCACAATGCCTTGCACATCTATATGAATGGAACAATGTCCCAGGCAGGGATCTGCC
AACGATCCTATCTTCCTTCTTCACCATGCATTTGTTGACAGTATTTTTGAGCAGTGGCTCCGAAGGCACCG
TCCTCTTCAAGAAGTTTATCCAGAAGCCAATGCACCCATTGGACATAACCGGGAATCCTACATGGTTCTTA
TACCACTGTACAGAAATGGTGATTTCTTTATTTCATCCAAAGATCTGGGCTATGACTATAGCTATCTACAA
GATTCAGACCCAGACTCTTTTCAAGACTACATTAAGTCCTATTTGGAACAAGCGAGTCGGATCTGGTCATG
GCTCCTTGGGGCGGCGATGGTAGGGGCCGTCCTCACTGCCCTGCTGGCGGGCTTGTGAGCTTGCTGTGTCC
TCACAAGAGAAAGCAGCTTCCTGAAGAAAAGCAGCCACTCCTCATGGAGAAAGAGGATTACCACAGCTTGT
ATCAGAGCCATTTATAAAAGGCTTAGGCAATAGAGTAGGGCCAAAAAGCCTGACCTCACTCTAACTCAAAG
TAATGTCCAGGTTCCCAGAGAATATCTGCTGGTATTTTTCTGTAAAGACCATTTGCAAAATTGTAACCTAA
TACAAAGTGTAGCCTTCTTCCAACTCAGGTAGAACACACCTGTCTTTGTCTTGCTGTTTTCACTCAGCCCT
TTTAACATTTTCCCCTAAGCCCATATGTCTAAGGAAAGGATGCTATTTGGTAATGAGGAACTGTTATTTGT
ATGTGAATTAAAGTGCTCTTATTTTAAAAAATTGAAATAATTTTGATTTTTGCCTTCTGATTATTTAAAGA
TCTATATATGTTTTATTGGCCCCTTCTTTATTTTAATAAAACAGTGAGAAATCT

With the help of the program Webcutter 2.0 use the appropriate restriction endonuclease to cut the sequence in such a way that you get a product 596 bp. Also, find which restriction enzymes cut the above sequence in 4 positions.

First we need to identify if given sequence is linear or circular. We run the relevant blastn 1.1 and we observe that given sequence has 100% query coverage with **Homo sapiens tyrosinase (TYR), mRNA**. This sequence, identified as Homo sapiens tyrosinase (TYR) mRNA, is **linear**. They are transcribed from DNA in the nucleus and then translated into proteins in the cytoplasm. Tyrosinase is an enzyme critical in the melanin biosynthesis pathway, involved in determining the color of the skin, hair, and eyes. In the context of molecular biology, mRNA molecules are synthesized from DNA templates during transcription. After transcription, mRNA undergoes processing events, including splicing, 5' capping, and 3' polyadenylation, resulting in a mature linear mRNA molecule that exits the nucleus to be translated into protein by ribosomes in the cytoplasm. Therefore, the Homo sapiens tyrosinase (TYR) mRNA, like all other eukaryotic mRNAs, is linear.

To perform analysis, we have to paste the sequence into the provided box. We then select All enzymes in the database on "Please indicate which enzymes to include in the analysis" setting and click the Analyze sequence button as depicted in figure 1.2. We perform a linear sequence analysis. Searching for restriction enzymes that cut the sequence into 4 positions can be performed more effectively if we select in "Please indicate which enzymes to include in the display" setting Enzymes cut exactly 4 times, so we do not have to search along all results depicted in figure 1.3.

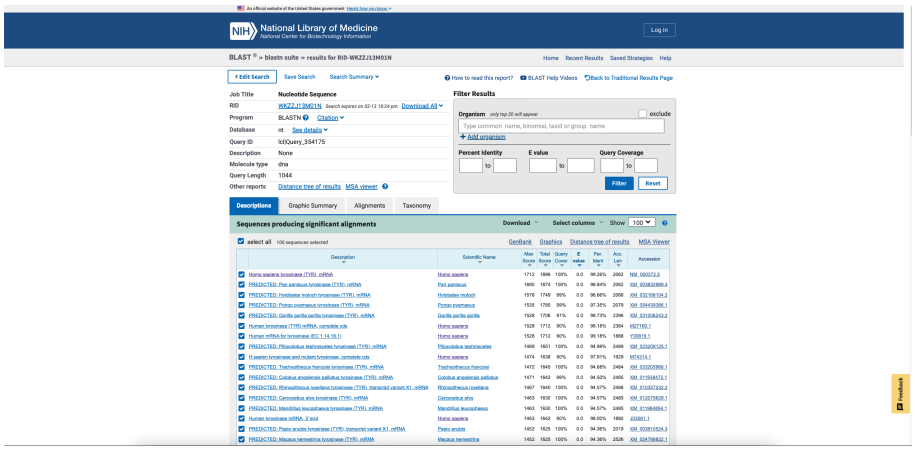

Figure 1.1. *blastn Sequence Result*

According to performed analysis 1.1, 1.2, 1.3, 1.4 , restriction endonuclease **BglII** cuts the sequence in such a way resulting in a product of 596 bp. A python script was written based on analysis performed on Heimanlab Webcutter 2.0 tool to identify which restriction endonuclease results in the desired product size base pairs.

```
# Author: Stamos Evangelos
# Date: 2024-02-02
# Title: bio1100 | Translational bioinformatics | Assignment | Part 1
# Description: This script calculates the length of a given sequence and identifies which endonuclease
    produces a fragment of 596 bp.

sequence = "CGAGGGACCTTTACGGCGTAATCCTGGAAACCATGACAAATCCAGAACCCCAAGGCTCCCCTCTTCAGCTGATGT
            AGAATTTTGCCTGAGTTTGACCCATGGAAGGATTTGCTAGTCCACTTACTGGGATAGCGGATGCCTCTCAAAGAG
            CATGCACAATGCCTTGCAATCTATATGAATGGAACAATGTCCCAGGCAGGGATCTGCCAACGATCCTATCTTCCT
```

```
            TCTTCACCATGCATTTGTTGACAGTATTTTTGAGCAGTGGCTCCGAAGGCACCGTCCTCTTCAAGAAGTTTATCC
            AGAAGCCAATGCACCCATTGGACATAACCGGGAATCCTACATGGTTCTTATACCACTGTACAGAAATGGTGATTT
            CTTTATTTCATCCAAAGATCTGGGCTATGACTATAGCTATCTACAAGATTCAGACCCAGACTCTTTTCAAGACTA
            CATTAAGTCCTATTTGGAACAAGCGAGTCGGATCTGGTCATGGCTCCTTGGGGCGGCGATGGTAGGGGCCGTCCT
            CACTGCCCTGCTGGCGGGCTTGTGAGCTTGCTGTGTCGTCACAAGAGAAAGCAGCTTCCTGAAGAAAAGCAGCCA
            CTCCTCATGGAGAAAGAGGATTACCACAGCTTGTATCAGAGCCATTTATAAAAGGCTTAGGCAATAGAGTAGGGC
            CAAAAAGCCTGACCTCACTCTAACTCAAAGTAATGTCCAGGTTCCCAGAGAATATCTGCTGGTATTTTTCTGTAA
            AGACCATTTGCAAAATTGTAACCTAATACAAAGTGTAGCCTTCTTCCAACTCAGGTAGAACACACCTGTCTTTGT
            CTTGCTGTTTTCACTCAGCCCTTTTAACATTTTCCCCTAAGCCCATATGTCTAAGGAAAGGATGCTATTTGGTAA
            TGAGGAACTGTTATTTGTATGTGAATTAAAGTGCTCTTATTTTAAAAAATTGAAATAATTTTGATTTTTGCCTTC
            TGATTATTTAAAGATCTATATATGTTTTATTGGCCCCTTCTTTATTTTAATAAAACAGTGAGAAATCT"


# Calculate the length of the DNA sequence
sequence_length = len(sequence)

# Print the length
print("Length of given sequence:", sequence_length, "bp")

# Based on analysis performed on the sequence, the following endonucleases and their cut positions are provided.
# Analysis was performed using the Heimanlab Webcutter 2.0 tool (http://heimanlab.com/cut2.html) with the following settings:
#   Type of analysis: Linear sequence analysis
#   Restriction sites displayed: Map of restriction sites
#                                Table of sites, sorted alphabetically by enzyme name
#   Displayed enzymes : All enzymes
#   Enzymes to include in the analysis: All enzymes in the database

# Given the provided list of endonucleases and their cut positions, identify which one produces a fragment of 596 bp.

# Defining the endonucleases with their cut positions
endonucleases = {
    "AccB1I": [274], "AccB7I": [123], "AciI": [135, 507, 543], "AclWI": [205, 217, 485],
    "AcsI": [77], "AfaI": [360], "AluI": [68, 412, 552, 580, 630], "Alw21I": [936],
    "AlwI": [205, 217, 485], "ApoI": [77], "AspHI": [936], "AspS9I": [5, 517, 673, 1008],
    "AsuI": [5, 517, 673, 1008], "AvaII": [5], "BanI": [274], "BbuI": [154],
    "Bbv12I": [936], "BbvI": [581, 599], "BcnI": [330], "BfaI": [112], "BglI": [537],
    "BglII": [392, 988], "Bme18I": [5], "BmyI": [936], "BsaJI": [50, 97, 192, 497],
    "Bsc4I": [101, 122, 197, 502, 538], "Bse1I": [127], "BseDI": [50, 97, 192, 497],
    "BseNI": [127], "BseRI": [607], "BshNI": [274], "BsiHKAI": [936], "BsiSI": [329],
    "BsiYI": [102, 123, 198, 503, 539], "BslI": [102, 123, 198, 503, 539], "BsmFI": [8, 194],
    "BsoFI": [505, 578, 596], "Bsp1286I": [936], "Bsp1407I": [358], "Bsp143I": [201, 212, 392, 481, 988],
    "Bsp19I": [97], "BsrGI": [358], "BsrI": [127], "BsrSI": [127], "BssT1I": [50, 97, 497],
    "Bst2UI": [24, 194, 714], "Bst71I": [581, 599], "BstDEI": [86, 657, 801, 840, 863, 877],
    "BstDSI": [97], "BstF5I": [138, 389, 890], "BstNI": [24, 194, 714], "BstOI": [24, 194, 714],
    "BstSFI": [407], "BstX2I": [201, 392, 481, 988], "BstXI": [395], "BstYI": [201, 392, 481, 988],
    "BsuRI": [519, 675, 1009], "Cac8I": [152, 538, 554], "Cfr13I": [5, 517, 673, 1008],
    "Csp6I": [359], "CviJI": [55, 68, 266, 306, 400, 412, 494, 519, 544, 552, 580, 598, 630, 642, 656, 675, 683, 789, 844, 867, 1009],
    "DdeI": [86, 657, 801, 840, 863, 877], "DpnI": [203, 214, 394, 483, 990], "DpnII": [201, 212, 392,
```

```
    481, 988],
    "DraI": [944, 985], "DraII": [5], "DsaI": [97], "Eam1104I": [66, 288], "EarI": [66, 288],
    "Eco130I": [50, 97, 497], "Eco47I": [5], "Eco57I": [68, 590], "Eco64I": [274], "EcoO109I": [5],
    "EcoRII": [22, 192,712],"EcoT14I": [50, 97, 497], "EcoT22I": [239], "ErhI": [50, 97, 497],
    "Esp1396I": [123], "FauI": [544], "FauNDI": [871],
    "FokI": [138, 389, 890], "Fsp4HI": [505, 578, 596],
    "HaeIII": [519, 675, 1009], "HapII": [329], "HgiEI": [5],
    "HincII": [245], "HindII": [245], "HinfI": [333, 423, 435, 476],
    "HpaII": [329], "HphI": [234, 373], "Hsp92II": [35, 101, 154, 237, 344, 493, 610],
    "ItaI": [505, 578, 596], "Ksp632I": [66, 288], "Kzo9I": [201, 212, 392, 481, 988],
    "MaeI": [112], "MaeIII": [563, 768], "MboI": [201, 212, 392, 481, 988],
    "MboII": [66, 224, 231, 288, 591, 797], "MflI": [201, 392, 481, 988],
    "MnlI": [5, 63, 142, 285, 527, 607, 620, 692, 906], "Mph1103I": [239],
    "MseI": [454, 850, 927, 943, 984, 1023], "MslI": [173], "MspA1I": [68],
    "MspI": [329], "MspR9I": [24, 194, 330, 714], "MvaI": [24, 194, 714],
    "MwoI": [272, 537], "NciI": [330], "NcoI": [97], "NdeI": [871],
    "NdeII": [201, 212, 392, 481, 988], "NlaIII": [35, 101, 154, 237, 344, 493, 610],
    "NlaIV": [6, 56, 267, 276, 495, 518, 718, 1010], "NsiI": [239], "NspBII": [68],
    "NspI": [154], "PaeI": [154], "PalI": [519, 675, 1009], "PflMI": [123],
    "PleI": [439, 480], "Ppu10I": [235], "PpuMI": [5], "Psp5II": [5],
    "PspN4I": [6, 56, 267, 276, 495, 518, 718, 1010], "PvuII": [68], "RsaI": [360],
    "Sau3AI": [201, 212, 392, 481, 988], "Sau96I": [5, 517, 673, 1008],
    "ScrFI": [24, 194, 330, 714], "SduI": [936], "SfaNI": [139, 891], "SfcI": [407],
    "SinI": [5], "SphI": [154], "Sse9I": [77, 764, 924, 948, 957], "SspBI": [358],
    "StyI": [50, 97, 497], "TfiI": [333, 423], "Tru1I": [454, 850, 927, 943, 984, 1023],
    "Tru9I": [454, 850, 927, 943, 984, 1023], "Tsp45I": [563], "Tsp509I": [77, 764, 924, 948, 957],
    "TspEI": [77, 764, 924, 948, 957],"TspRI": [265, 359, 531, 1036], "XhoII": [201, 392, 481, 988],
    "Zsp2I": [239]}


# Function to calculate fragment sizes
def calculate_fragment_sizes(cut_positions, sequence_length):
    # Add the start and end of the sequence to the list of cut positions
    cut_positions = [0] + sorted(cut_positions) + [sequence_length]
    # Calculate the sizes of the resulting fragments
    fragment_sizes = [cut_positions[i+1] - cut_positions[i] for i in range(len(cut_positions)-1)]
    return fragment_sizes

# Check each endonuclease for a 596 bp fragment
enzyme_with_596bp = {}

for enzyme, positions in endonucleases.items():
    sizes = calculate_fragment_sizes(positions, sequence_length)
    if 596 in sizes:
        enzyme_with_596bp[enzyme] = sizes

if not enzyme_with_596bp:  # Zero endonucleases found that produce a 596 bp fragment
    print('No endunucleases found that produce a 596 bp fragment.')
else:   # Print the endonuclease(s) that produce a 596 bp fragment
    print('Endonuclease(s) that cut the sequence resulting to a product of 596 base pairs:',
     enzyme_with_596bp)

```

```
# Produced result
# 'Length of given sequence: 1044 bp'
#'Endonuclease(s) that cut the sequence resulting to a product of 596 base pairs: {'BglII': [392, 596,
    56]}'
```

Listing 1.1: Exercise 1 - Python Code

Restriction enzymes **AspS9I, AsuI, BsaJI, BseDI, BstX2I, BstYI, Cfr13I, HinfI, MflI, MspR9I, Sau96I, ScrFI, TspRI** and **XhoII** cut the given sequence in 4 positions, with further information available in table 1.5.

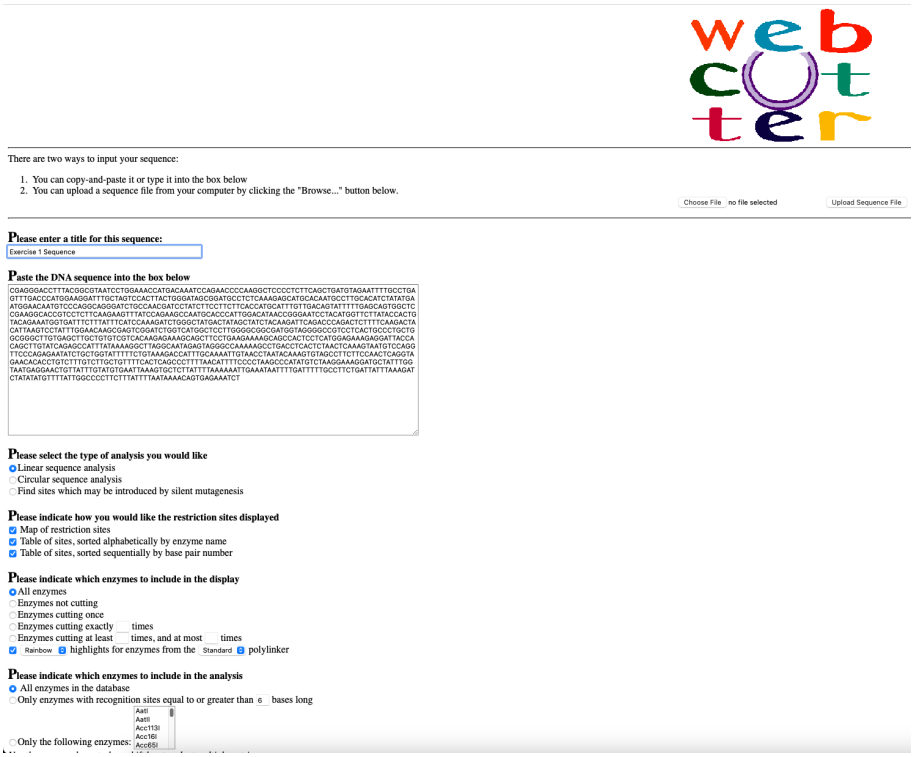

Figure 1.2. *Webcutter 2.0 Settings*

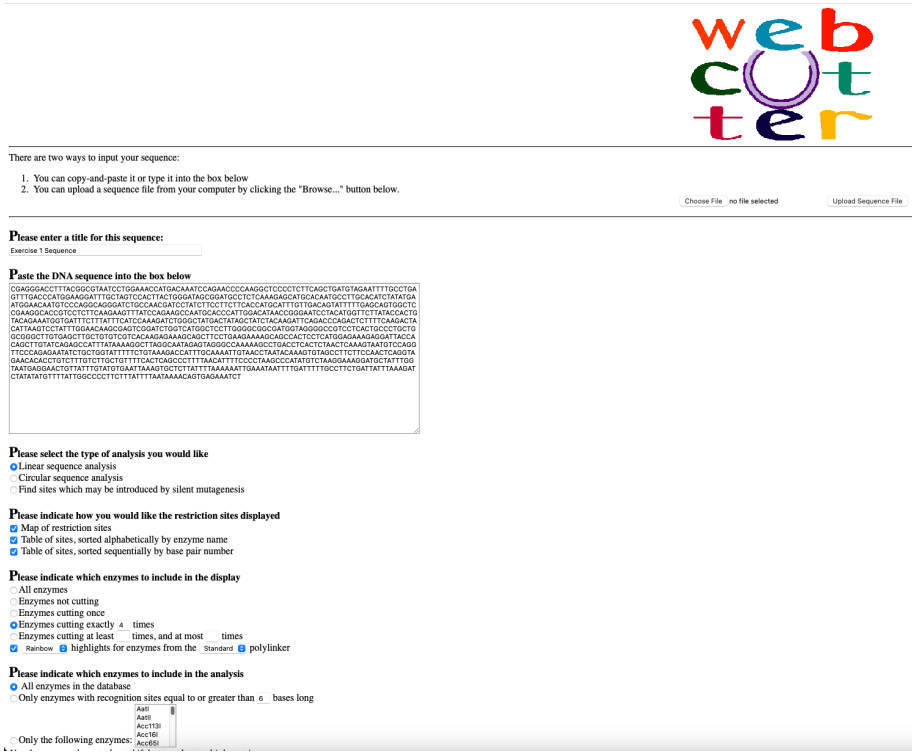

Figure 1.3. *Webcutter 2.0 Settings for restriction enzymes that cut the sequence in 4 positions*

Table 1.1. *Endonucleases that do not cut the sequence*

Endonucleases	**Endonucleases (continue)**
AatI, AatII, Acc113I, Acc16I, Acc65I, AccBSI, AccI, AccII, AccIII, AclNI, AcyI, AfeI, AflII, AflIII, AgeI, AhdI, Alw26I, Alw44I, AlwNI, Ama87I, AocI, Aor51HI, ApaI, ApaLI, AscI, AseI, AsnI, Asp700I, Asp718I, AspEI, AspI, AspLEI, AtsI, AvaI, AviII, AvrII, BalI, BamHI, BanII, BanIII, BbeI, BbiII, BbrPI, BbsI, Bbv16II, BcgI, BclI, BcoI, BfrI, BlnI, BlpI, BpiI, BpmI, Bpu1102I, Bpu14I, BpuAI, Bsa29I, BsaAI, BsaBI, BsaHI, BsaI, BsaMI, BsaOI, BsaWI, BscI, Bse118I, Bse21I, Bse8I, BseAI, BseCI, BsePI, BsgI, Bsh1236I, Bsh1285I, Bsh1365I, BsiEI, BsiI, BsiMI, BsiWI, BsmAI, BsmBI, BsmI, BsoBI, Bsp106I, Bsp119I, Bsp120I, Bsp13I, Bsp143II, Bsp1720I, Bsp68I, BspCI, BspDI, BspEI, BspHI, BspLU11I, BspMI, BspTI, BspXI, BsrBI, BsrBRI, BsrDI, BsrFI, BssAI, BssHII, BssSI, Bst1107I, Bst98I, BstBI, BstD102I, BstEII, BstH2I, BstI, BstMCI, BstPI, BstSNI, BstUI, BstZI, Bsu15I, Bsu36I, CciNI, CelII, CfoI, Cfr10I, Cfr42I, Cfr9I, CfrI, ClaI, CpoI, Csp45I, CspI, CvnI, DraIII, DrdI	EaeI, EagI, Eam1105I, Ecl136II, EclHKI, EclXI, Eco105I, Eco147I, Eco24I, Eco255I, Eco31I, Eco32I, Eco47III, Eco52I, Eco72I, Eco81I, Eco88I, Eco91I, EcoICRI, EcoNI, EcoO65I, EcoRI, EcoRV, EheI, Esp3I, FbaI, FriOI, FseI, FspI, GsuI, HaeII, HgaI, HhaI, Hin1I, Hin6I, HinP1I, HindIII, HpaI, Hsp92I, HspAI, KasI, Kpn2I, KpnI, Ksp22I, KspI, LspI, MaeII, MamI, MfeI, MluI, MluNI, MroI, MroNI, MscI, Msp17I, MspCI, MunI, Mva1269I, MvnI, NaeI, NarI, NgoAIV, NgoMI, NheI, NotI, NruI, NspV, PacI, PaeR7I, Pfl23II, PinAI, Ple19I, PmaCI, Pme55I, PmeI, PmlI, PshAI, PshBI, Psp124BI, Psp1406I, PspAI, PspALI, PspEI, PspLI, PspOMI, PstI, PstNHI, PvuI, RcaI, RsrII, SacI, SacII, SalI, SapI, SbfI, ScaI, SexAI, SfiI, Sfr274I, Sfr303I, SfuI, SgfI, SgrAI, SmaI, SmiI, SnaBI, SpeI, SplI, SrfI, Sse8387I, SseBI, SspI, SstI, SstII, StuI, SunI, SwaI, TaqI, ThaI, Tth111I, TthHB8I, Vha464I, VneI, VspI, XbaI, XcmI, XhoI, XmaI, XmaIII, XmnI

Table 1.2. *Table of restriction enzymes 1 of 3*

Enzyme name	No. of cuts	Positions of sites	Recognition sequence
AccB1I	1	274	g/gyrcc
AccB7I	1	123	ccannnn/ntgg
AciI	3	135 507 543	ccgc
AclWI	3	205 217 485	ggatc
AcsI	1	77	r/aatty
AfaI	1	360	gt/ac
AluI	5	68 412 552 580 630	ag/ct
Alw21I	1	936	gwgcw/c
AlwI	3	205 217 485	ggatc
ApoI	1	77	r/aatty
AspHI	1	936	gwgcw/c
AspS9I	4	5 517 673 1008	g/gncc
AsuI	4	5 517 673 1008	g/gncc
AvaII	1	5	g/gwcc
BanI	1	274	g/gyrcc
BbuI	1	154	gcatg/c
Bbv12I	1	936	gwgcw/c
BbvI	2	581 599	gcagc
BcnI	1	330	cc/sgg
BfaI	1	112	c/tag
BglI	1	537	gccnnnn/nggc
BglII	**2**	**392 988**	**a/gatct**
Bme18I	1	5	g/gwcc
BmyI	1	936	gdgch/c
BsaJI	4	50 97 192 497	c/cnngg
Bsc4I	5	101 122 197 502 538	ccnnnn/nnngg
Bse1I	1	127	actgg
BseDI	4	50 97 192 497	c/cnngg
BseNI	1	127	actgg
BseRI	1	607	gaggag
BshNI	1	274	g/gyrcc
BsiHKAI	1	936	gwgcw/c
BsiSI	1	329	c/cgg
BsiYI	5	102 123 198 503 539	ccnnnnn/nngg
BslI	5	102 123 198 503 539	ccnnnnn/nngg
BsmFI	2	8 194	gggac
BsoFI	3	505 578 596	gc/ngc
Bsp1286I	1	936	gdgch/c
Bsp1407I	1	358	t/gtaca
Bsp143I	5	201 212 392 481 988	/gatc
Bsp19I	1	97	c/catgg
BsrGI	1	358	t/gtaca
BsrI	1	127	actgg
BsrSI	1	127	actgg
BssT1I	3	50 97 497	c/cwwgg
Bst2UI	3	24 194 714	cc/wgg
Bst71I	2	581 599	gcagc
BstDEI	6	86 657 801 840 863 877	c/tnag
BstDSI	1	97	c/crygg
BstF5I	3	138 389 890	ggatg
BstNI	3	24 194 714	cc/wgg
BstOI	3	24 194 714	cc/wgg
BstSFI	1	407	c/tryag
BstX2I	4	201 392 481 988	r/gatcy
BstXI	1	395	ccannnnn/ntgg
BstYI	4	201 392 481 988	r/gatcy

Table 1.3. *Table of restriction enzymes 2 of 3*

Enzyme name	No. of cuts	Positions of sites	Recognition sequence
BsuRI	3	519 675 1009	gg/cc
Cac8I	3	152 538 554	gcn/ngc
Cfr13I	4	5 517 673 1008	g/gncc
Csp6I	1	359	g/tac
CviJI	21	55 68 266 306 400 412 494 519 544 552 580 598 630 642 656 675 683 789 844 867 1009	rg/cy
DdeI	6	86 657 801 840 863 877	c/tnag
DpnI	5	203 214 394 483 990	ga/tc
DpnII	5	201 212 392 481 988	/gatc
DraI	2	944 985	ttt/aaa
DraII	1	5	rg/gnccy
DsaI	1	97	c/crygg
Eam1104I	2	66 288	ctcttc
EarI	2	66 288	ctcttc
Eco130I	3	50 97 497	c/cwwgg
Eco47I	1	5	g/gwcc
Eco57I	2	68 590	ctgaag
Eco64I	1	274	g/gyrcc
EcoO109I	1	5	rg/gnccy
EcoRII	3	22 192 712	/ccwgg
EcoT14I	3	50 97 497	c/cwwgg
EcoT22I	1	239	atgca/t
ErhI	3	50 97 497	c/cwwgg
Esp1396I	1	123	ccannnn/ntgg
FauI	1	544	cccgc
FauNDI	1	871	ca/tatg
FokI	3	138 389 890	ggatg
Fsp4HI	3	505 578 596	gc/ngc
HaeIII	3	519 675 1009	gg/cc
HapII	1	329	c/cgg
HgiEI	1	5	g/gwcc
HincII	1	245	gty/rac
HindII	1	245	gty/rac
HinfI	4	333 423 435 476	g/antc
HpaII	1	329	c/cgg
HphI	2	234 373	ggtga
Hsp92II	7	35 101 154 237 344 493 610	catg/
ItaI	3	505 578 596	gc/ngc
Ksp632I	2	66 288	ctcttc
Kzo9I	5	201 212 392 481 988	/gatc
MaeI	1	112	c/tag
MaeIII	2	563 768	/gtnac
MboI	5	201 212 392 481 988	/gatc
MboII	6	66 224 231 288 591 797	gaaga
MflI	4	201 392 481 988	r/gatcy
MnlI	9	5 63 142 285 527 607 620 692 906	cctc
Mph1103I	1	239	atgca/t
MseI	6	454 850 927 943 984 1023	t/taa
MslI	1	173	caynn/nnrtg
MspA1I	1	68	cmg/ckg
MspI	1	329	c/cgg
MspR9I	4	24 194 330 714	cc/ngg
MvaI	3	24 194 714	cc/wgg
MwoI	2	272 537	gcnnnnn/nngc

Table 1.4. *Table of restriction enzymes 3 of 3*

Enzyme name	No. of cuts	Positions of sites	Recognition sequence
NciI	1	330	cc/sgg
NcoI	1	97	c/catgg
NdeI	1	871	ca/tatg
NdeII	5	201 212 392 481 988	/gatc
NlaIII	7	35 101 154 237 344 493 610	catg/
NlaIV	8	6 56 267 276 495 518 718 1010	ggn/ncc
NsiI	1	239	atgca/t
NspBII	1	68	cmg/ckg
NspI	1	154	rcatg/y
PaeI	1	154	gcatg/c
PalI	3	519 675 1009	gg/cc
PflMI	1	123	ccannnn/ntgg
PleI	2	439 480	gagtc
Ppu10I	1	235	a/tgcat
PpuMI	1	5	rg/gwccy
Psp5II	1	5	rg/gwccy
PspN4I	8	6 56 267 276 495 518 718 1010	ggn/ncc
PvuII	1	68	cag/ctg
RsaI	1	360	gt/ac
Sau3AI	5	201 212 392 481 988	/gatc
Sau96I	4	5 517 673 1008	g/gncc
ScrFI	4	24 194 330 714	cc/ngg
SduI	1	936	gdgch/c
SfaNI	2	139 891	gcatc
SfcI	1	407	c/tryag
SinI	1	5	g/gwcc
SphI	1	154	gcatg/c
Sse9I	5	77 764 924 948 957	/aatt
SspBI	1	358	t/gtaca
StyI	3	50 97 497	c/cwwgg
TfiI	2	333 423	g/awtc
Tru1I	6	454 850 927 943 984 1023	t/taa
Tru9I	6	454 850 927 943 984 1023	t/taa
Tsp45I	1	563	/gtsac
Tsp509I	5	77 764 924 948 957	/aatt
TspEI	5	77 764 924 948 957	/aatt
TspRI	4	265 359 531 1036	cagtg
Van91I	1	123	ccannnn/ntgg
XhoII	4	201 392 481 988	r/gatcy
Zsp2I	1	239	atgca/t

Table 1.5. *Table of restriction enzymes that cut sequence in 4 positions*

Enzyme name	No. of cuts	Position site	Recognition sequence
AspS9I	4	5 517 673 1008	g/gncc
AsuI	4	5 517 673 1008	g/gncc
BsaJI	4	50 97 192 497	c/cnngg
BseDI	4	50 97 192 497	c/cnngg
BstX2I	4	201 392 481 988	r/gatcy
BstYI	4	201 392 481 988	r/gatcy
Cfr13I	4	5 517 673 1008	g/gncc
HinfI	4	333 423 435 476	g/antc
MflI	4	201 392 481 988	r/gatcy
MspR9I	4	24 194 330 714	cc/ngg
Sau96I	4	5 517 673 1008	g/gncc
ScrFI	4	24 194 330 714	cc/ngg
TspRI	4	265 359 531 1036	cagtg
XhoII	4	201 392 481 988	r/gatcy

Chapter 2

Exercise 2

For the 596 bp product you got in Exercise 1, select primers using any program you want to carry out the polymerase chain reaction (PCR). Select the pair of primers that gives you the largest product.

In Exercise 1, we observed that using restrictive endonuclease **BglII** results in a product of 596 base pairs. Given that BglII cuts the linear sequence at positions 392 and 988, and considering the length of our sequence is 1044 base pairs, the digestion with BglII produces the following fragments:

- Fragment 1: This fragment is from the start of the sequence to the first cut site at position 392, 392 base pairs long.
- Fragment 2: This fragment is between the two cut sites, from position 392 to 988, 596 base pairs long.
- Fragment 3: This fragment is from the second cut site at position 988 to the end of the sequence at position 1.044, 56 base pairs long.

We are interested in **Fragment 2**, which starts from position 392 of the given sequence to position 988.

An alternative way of finding the desired cut sequence is implemented in a Python script.

```
# Author: Stamos Evangelos
# Date: 2024-02-02
# Title: bio1100 | Translational bioinformatics | Assignment | Part 2
# Description: This script calculates the fragment of a sequence given cut positions.

sequence = "CGAGGGACCTTTACGGCGTAATCCTGGAAACCATGACAAATCCAGAACCCCAAGGCTCCCCTCTTCAGCTGATGT
            AGAATTTTGCCTGAGTTTGACCCATGGAAGGATTTGCTAGTCCACTTACTGGGATAGCGGATGCCTCTCAAAGAG
            CATGCACAATGCCTTGCAATCTATATGAATGGAACAATGTCCCAGGCAGGGATCTGCCAACGATCCTATCTTCCT
            TCTTCACCATGCATTTGTTGACAGTATTTTTGAGCAGTGGCTCCGAAGGCACCGTCCTCTTCAAGAAGTTTATCC
            AGAAGCCAATGCACCCATTGGACATAACCGGGAATCCTACATGGTTCTTATACCACTGTACAGAAATGGTGATTT
            CTTTATTTCATCCAAAGATCTGGGCTATGACTATAGCTATCTACAAGATTCAGACCCAGACTCTTTTCAAGACTA
            CATTAAGTCCTATTTGGAACAAGCGAGTCGGATCTGGTCATGGCTCCTTGGGGCGGCGATGGTAGGGGCCGTCCT
            CACTGCCCTGCTGGCGGGCTTGTGAGCTTGCTGTGTCGTCACAAGAGAAAGCAGCTTCCTGAAGAAAAGCAGCCA
            CTCCTCATGGAGAAAGAGGATTACCACAGCTTGTATCAGAGCCATTTATAAAAGGCTTAGGCAATAGAGTAGGGC
            CAAAAAGCCTGACCTCACTCTAACTCAAAGTAATGTCCAGGTTCCCAGAGAATATCTGCTGGTATTTTTCTGTAA
            AGACCATTTGCAAAATTGTAACCTAATACAAAGTGTAGCCTTCTTCCAACTCAGGTAGAACACACCTGTCTTTGT
            CTTGCTGTTTTCACTCAGCCCTTTTAACATTTTCCCCTAAGCCCATATGTCTAAGGAAAGGATGCTATTTGGTAA
            TGAGGAACTGTTATTTGTATGTGAATTAAAGTGCTCTTATTTTAAAAAATTGAAATAATTTTGATTTTTGCCTTC
            TGATTATTTAAAGATCTATATATGTTTTATTGGCCCCTTCTTTATTTTAATAAAACAGTGAGAAATCT"

# Note 1st position is 0
fragment_596bp = sequence[391:987]

print("596 bp fragment:", fragment_596bp)

# Output
# 596 bp fragment:AGATCTGGGCTATGACTATAGCTATCTACAAGATTCAGACCCAGACTCTTTTCAAGACTACATTAAGTCC
TATTTGGAACAAGCGAGTCGGATCTGGTCATGGCTCCTTGGGGCGGCGATGGTAGGGGCCGTCCTCACTGCCCTGCTGGCGGGCTTG
TGAGCTTGCTGTGTCGTCACAAGAGAAAGCAGCTTCCTGAAGAAAAGCAGCCACTCCTCATGGAGAAAGAGGATTACCACAGCTTGT
ATCAGAGCCATTTATAAAAGGCTTAGGCAATAGAGTAGGGCCAAAAAGCCTGACCTCACTCTAACTCAAAGTAATGTCCAGGTTCCC
AGAGAATATCTGCTGGTATTTTTCTGTAAAGACCATTTGCAAAATTGTAACCTAATACAAAGTGTAGCCTTCTTCCAACTCAGGTAG
AACACACCTGTCTTTGTCTTGCTGTTTTCACTCAGCCCTTTTAACATTTTCCCCTAAGCCCATATGTCTAAGGAAAGGATGCTATTT
```

```
GGTAATGAGGAACTGTTATTTGTATGTGAATTAAAGTGCTCTTATTTTAAAAAATTGAAATAATTTTGATTTTTGCCTTCTGATTAT
TTAA
```

Listing 2.1: Exercise 2 - Python Code

Since we have obtained our sequence, we may proceed to find primers for PCR and select the pair of primers that results in the biggest longer product, using NCBI Primer-BLAST. We set Forward prime range From 392 and Reverse primer range to 988 to input the initial sequence 2.1. Alternatively, we can input the cut sequence produced by our Python script and not select any range for primers. As we can see in detailed primer reports figures 2.3, 2.4, 2.5 and at table 2.1 **Primer pair 7** with **Forward primer TAGCTATCTACAAGATTCAGACCCA** and **Reverse primer AGGGCTGAGTGAAAACAGCA** results in **the longer product of 439 base pairs length**.

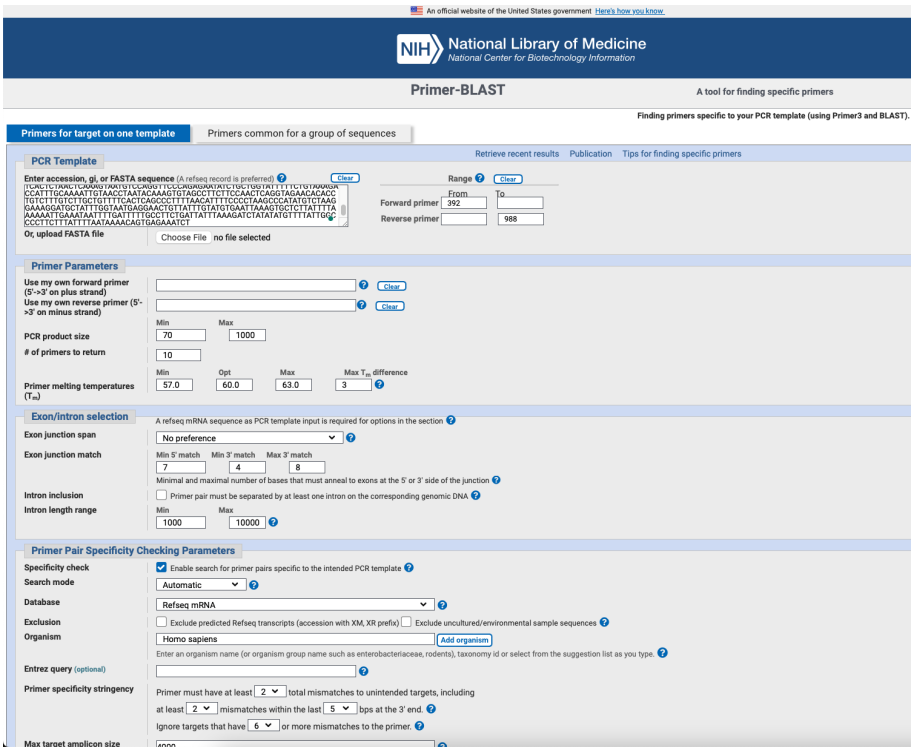

Figure 2.1. *NCBI Primer-Blast - Analysis settings*

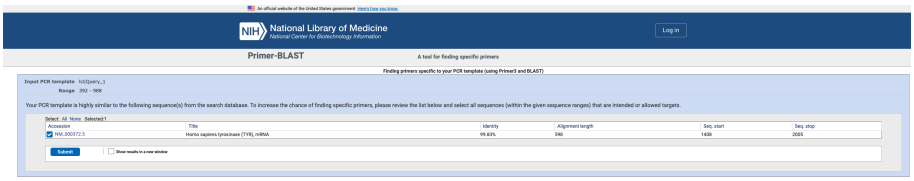

Figure 2.2. *NCBI Primer-Blast - PCR template highly similar to Homo sapiens tyrosinase (TYR) mRNA*

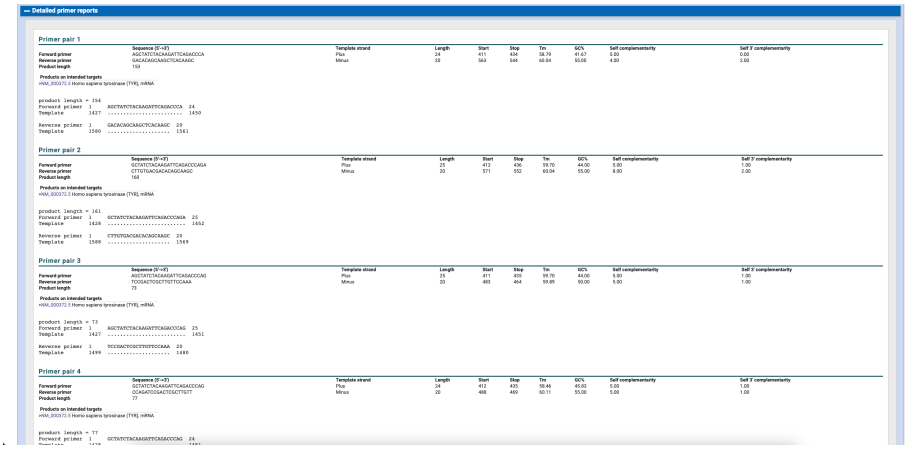

Figure 2.3. *NCBI Primer-Blast - Detailed primer reports I*

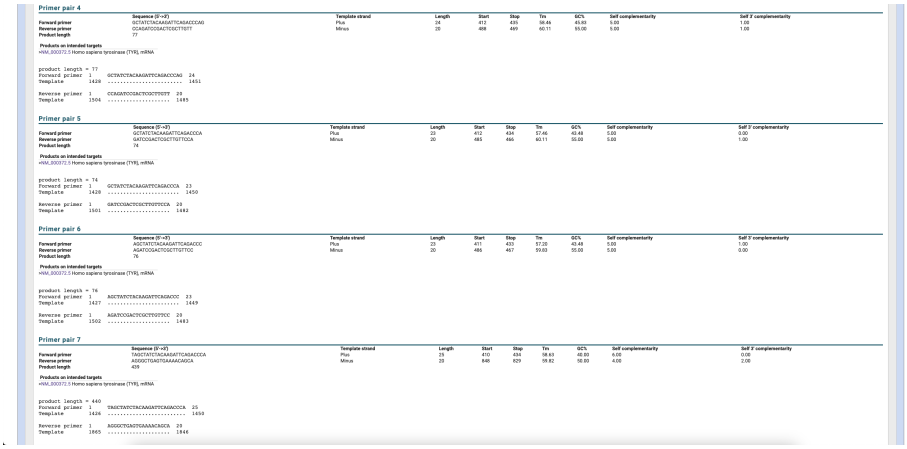

Figure 2.4. *NCBI Primer-Blast - Detailed primer reports II*

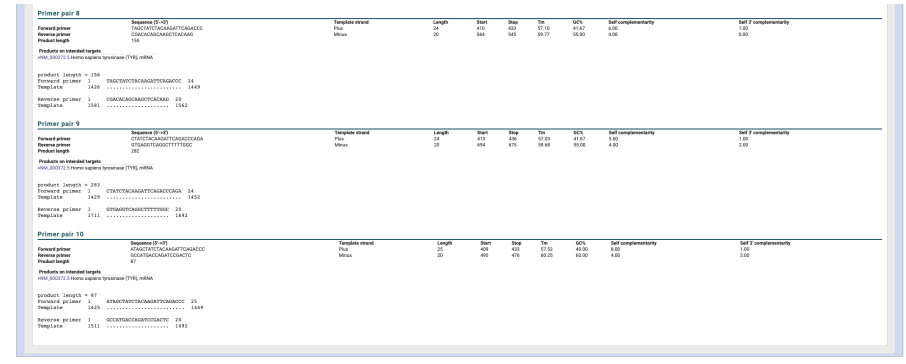

Figure 2.5. *NCBI Primer-Blast - Detailed primer reports II*

Table 2.1. *Primer pair and product size*

Forward Primer	Reverse Primer	Product size
AGCTATCTACAAGATTCAGACCCA	GACACAGCAAGCTCACAAGC	153
GCTATCTACAAGATTCAGACCCAGA	CTTGTGACGACACAGCAAGC	160
AGCTATCTACAAGATTCAGACCCAG	TCCGACTCGCTTGTTCCAAA	73
GCTATCTACAAGATTCAGACCCAG	CCAGATCCGACTCGCTTGTT	77
GCTATCTACAAGATTCAGACCCA	GATCCGACTCGCTTGTTCCA	74
AGCTATCTACAAGATTCAGACCC	AGATCCGACTCGCTTGTTCC	76
TAGCTATCTACAAGATTCAGACCCA	**AGGGCTGAGTGAAAACAGCA**	**439**
TAGCTATCTACAAGATTCAGACCC	CGACACAGCAAGCTCACAAG	155
CTATCTACAAGATTCAGACCCAGA	GTGAGGTCAGGCTTTTTGGC	282
ATAGCTATCTACAAGATTCAGACCC	GCCATGACCAGATCCGACTC	87

Chapter 3

Exercise 3

The mutations of the genes HEXA, HFE, PKU, PKU, HBB, LDLR cause respectively 5 well-known genetic diseases in humans. Which gene mutation causes β thalassemia?

Find for for this protein:

- Its sequence
- Its structure
- The post-translational modification
- The 4 amino acids that change and cause beta-thalassemia

Based on NBCI Gene search information obtained the following results are displayed in 3.1, 3.2, 3.3, 3.4, 3.5.

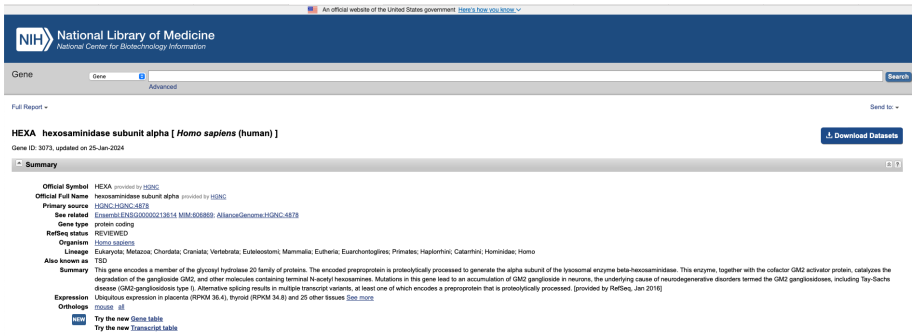

Figure 3.1. *NCBI Gene Information - HEXA*

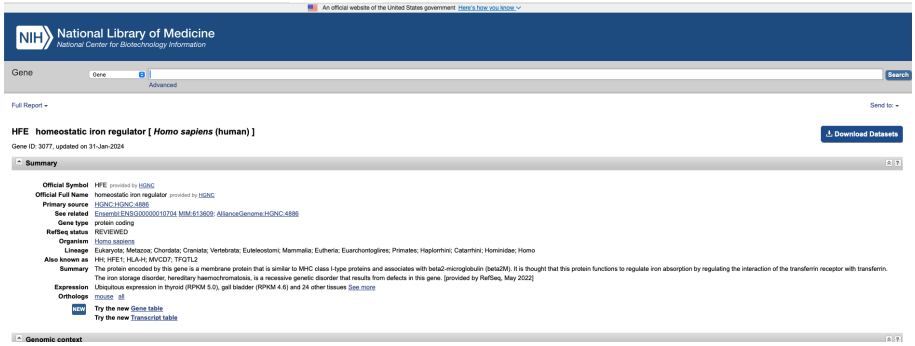

Figure 3.2. *NCBI Gene Information - HFE*

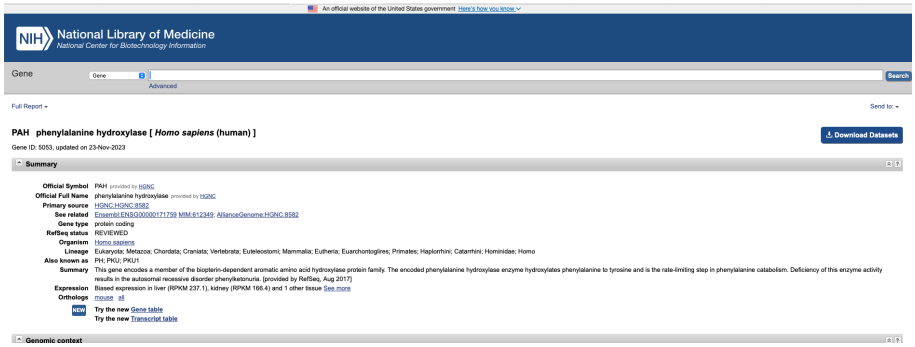

Figure 3.3. *NCBI Gene Information - PKU*

The mutations in the genes mentioned above are associated with the following diseases:

HEXA: Mutations in this gene cause Tay-Sachs disease, a rare inherited disorder that results in progressive destruction of nerve cells in the brain and spinal cord.

HFE: Mutations in the HFE gene cause Hemochromatosis, a condition of excess iron in the body that can lead to serious conditions such as diabetes, heart problems, and liver disease.

PKU: This is likely a reference to the PAH gene, which when mutated, causes Phenylketonuria (PKU). PKU is a genetic disorder that causes increased levels of phenylalanine (an

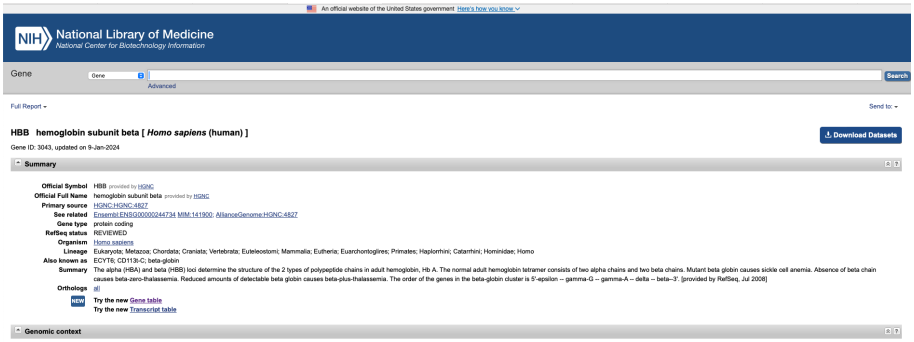

Figure 3.4. *NCBI Gene Information - HBB*

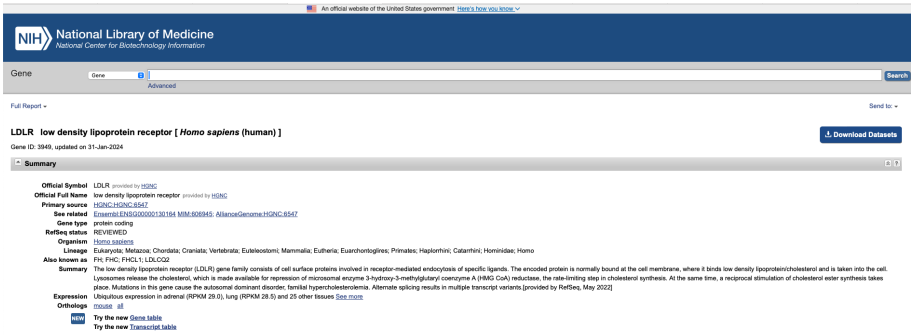

Figure 3.5. *NCBI Gene Information - LDLR*

amino acid) in the body.

HBB: **Mutations in the HBB gene cause β thalassemia.** The HBB gene provides instructions for making a protein called beta-globin, which is a component of hemoglobin. Hemoglobin is the protein in red blood cells that carries oxygen throughout the body. Beta thalassemia is a blood disorder that reduces the production of hemoglobin. Mutations in the HBB gene can lead to reduced (β+) or absent (β^0) production of beta-globin. This reduction or absence disrupts the normal balance of alpha-globin to beta-globin, leading to the formation of abnormal hemoglobin molecules that can destroy red blood cells, causing anemia and other related health issues. The severity of beta thalassemia can vary depending on the specific mutations in the HBB gene and how much they affect beta-globin production. The condition is inherited in an autosomal recessive pattern, meaning that two copies of the mutated gene (one from each parent) are necessary for a child to be affected by the disorder.

LDLR: Mutations in the LDLR gene cause Familial Hypercholesterolemia, a form of high cholesterol that can lead to serious health conditions like coronary artery disease.

Having identified that mutation on gene HBB causes β-thalassemia we proceed further on analysis using UniProt.

Table 3.1. *Genes mutation and diseases causality*

Gene Mutation	Caused Disease
HEXA	Tay-Sachs Disease
HFE	Hemochromatosis
PKU	Phenylketonuria
HBB	**β thalassemia**
LDLR	Hypercholesterolemia

3.1 Sequence

The sequence of HBB has length 147 and Mass (Da) 15,998, as depicted in 3.6 is the following:

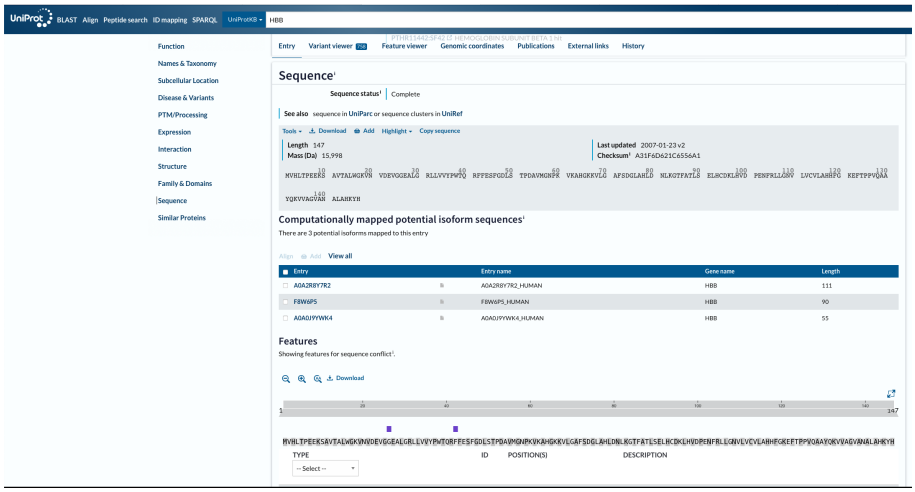

Figure 3.6. *UniProt - HBB Sequence*

```
MVHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPK
VKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFG
KEFTPPVQAAYQKVVAGVANALAHKYH
```

3.2 Structure

The structure of HBB is displayed in figure 3.7 and 3.8. It should be noted that displayed HBB structure is sourced by Protein Data Bank (PDB), with identifier 1A00, X-ray method, resolution 2.00 Å, chain B/D and positions 1 - 147. Many more structures are available on Uni-Prot, most of them of X-ray method while there are also 2 Neutron, 2 Nuclear Magnetic Resonance (NMR), 8 Electron Microscopy (EM) method structures from PDB and 1 predicted structure from AlphaFold 3.2. Structural Features of HBB are included in table 3.3.

3.3 Post-Translational Modification (PTM)

Post-translational modification (PTM) is the following:

Source	Identifier	Method	Chain	Positions
PDB	2DXM	Neutron	B/D	2-147
PDB	3KMF	Neutron	C/G	2-147
PDB	2H35	NMR	B/D	2-147
PDB	2M6Z	NMR	B/D	2-147
PDB	5NI1	EM	B/D	2-147
PDB	6NBC	EM	B/D	2-144
PDB	6NBD	EM	B/D	2-144
PDB	7PCF	EM	B	2-147
PDB	7PCH	EM	B/D	2-147
PDB	7PCQ	EM	B/D	2-147
PDB	7VDE	EM	B/D	1-147
PDB	7XGY	EM	B/D	1-147
AlphaFold	AF-P68871-F1	Predicted		1-147

Table 3.2. *Neutron, NMR, EM available HBB structures on UniProt*

Type	Positions	Sequence
Helix	6-17	PEEKSAVTALWG
Helix	21-35	VDEVGGEALGRLLVV
Helix	37-42	PWTQRF
Helix	44-46	ESF
Helix	52-57	PDAVMG
Helix	59-75	PKVKAHGKKVLGAFSDG
Turn	78-80	HLD
Helix	82-95	LKGTFATLSELHCD
Helix	102-119	ENFRLLGNVLVCVLAHHF
Helix	120-122	GKE
Helix	125-143	PPVQAAYQKVVAGVANALA
Helix	144-146	HKY

Table 3.3. *Structural Features of HBB*

Glucose reacts non-enzymatically with the N-terminus of the beta chain to form a stable ketoamine linkage. This takes place slowly and continuously throughout the 120-day life span of the red blood cell. The rate of glycation is increased in patients with diabetes mellitus.

S-nitrosylated; a nitric oxide group is first bound to Fe^{2+} and then transferred to Cys-94 to allow capture of O_2.

Acetylated on Lys-60, Lys-83 and Lys-145 upon aspirin exposure.

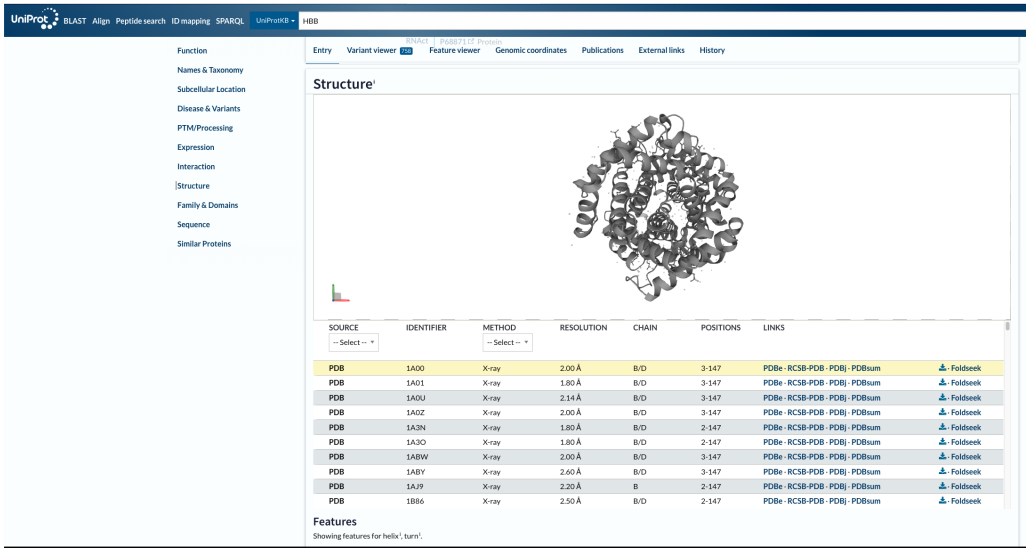

Figure 3.7. *UniProt - HBB Structure I*

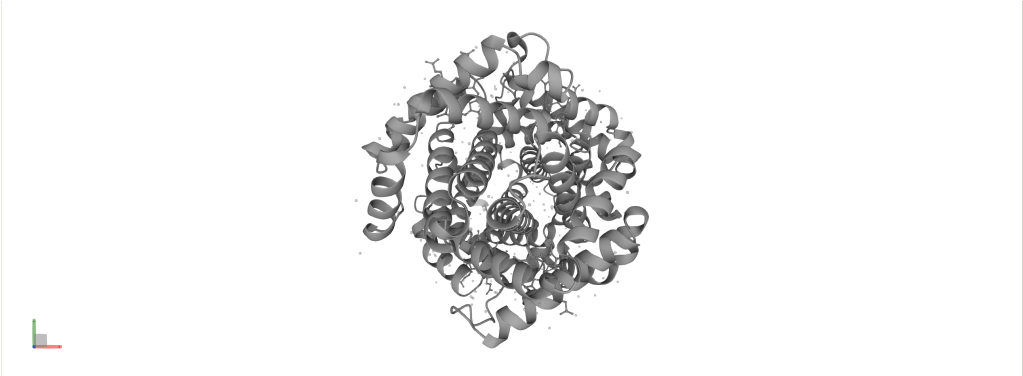

Figure 3.8. *UniProt - HBB Structure II*

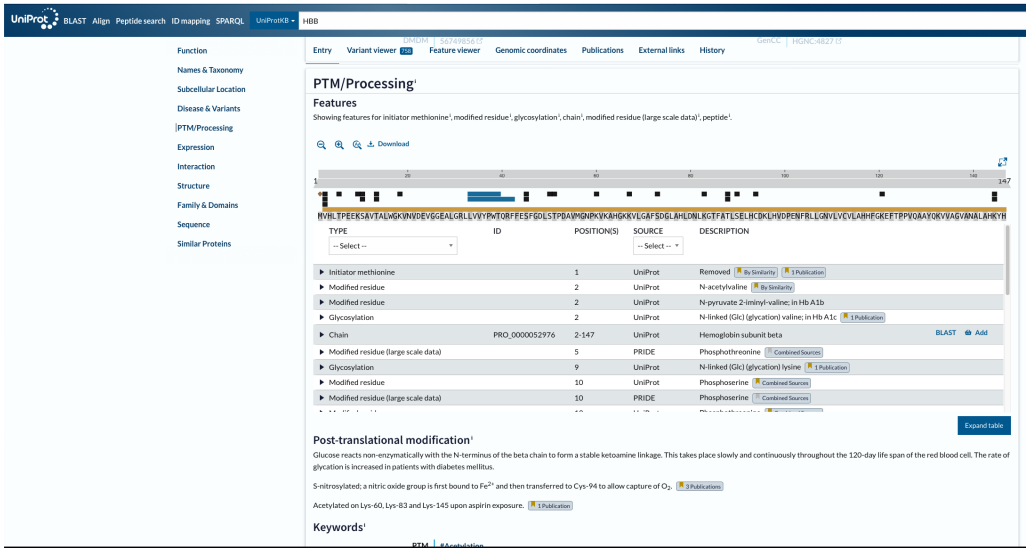

Figure 3.9. *UniProt - HBB PTM*

3.4 The 4 amino acids that change and cause beta-thalassemia

The four (4) amino acids that change and cause beta-thalassemia are **glutamic acid (E), lysine (K), leucine (L)** and **proline (P)**, as depicted in figure 3.10.

1. A mutation that causes a change from **glutamic acid (E)** to **lysine (K)** at position 27.
2. A mutation that leads to a change from **leucine (L)** to **proline (P)** at position 115.
3. A mutation that results in a change from **alanine (A)** to **aspartic acid (D)** at position 116.
4. A mutation that changes **valine (V)** to **glycine (G)** at position 127.

MVHLTPEEKSAVTALWGKVNVDEVGG**E**ALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPK
VKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCV**LA**HHFG
KEFTPP**V**QAAYQKVVAGVANALAHKYH

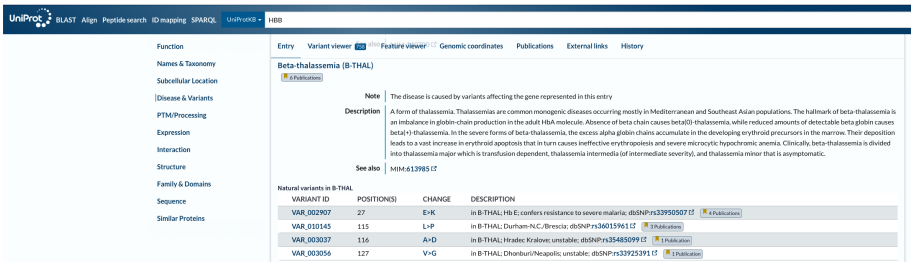

Figure 3.10. *UniProt - HBB The 4 amino acids that change and cause beta-thalassemia*

Chapter 4

Exercise 4

Using the beta thalassemia protein sequence, perform the relevant BLAST. Select the sequences Homo sapiens, Gorilla gorilla gorilla, Pongo abelii, Nomascus leukogenys, Trachypithecus francois, Pongo pygmaeus and using whichever program you wish, construct a Neighbor Joining phylogenetic tree. What do you observe?

4.1 NCBI Blast Neighbor Joining Phylogenetic Tree

Having obtained beta thalassaemia protein sequence (HBB Homo sapiens), we perform the relevant Protein BLAST blastp 4.1. We select the species for which we want to create a neighbor joining phylogenetic tree as shown in figures 4.2, 4.3, 4.4 and we save their sequences to perform a separate analysis with MEGA11, and we select 'Distance tree of results' to display it, choosing 'Neighbor Joining' in Tree Method option. We select 'Taxonomic Name (Sequence ID)' in Sequence Label option for a more clear visualized result. 4.5

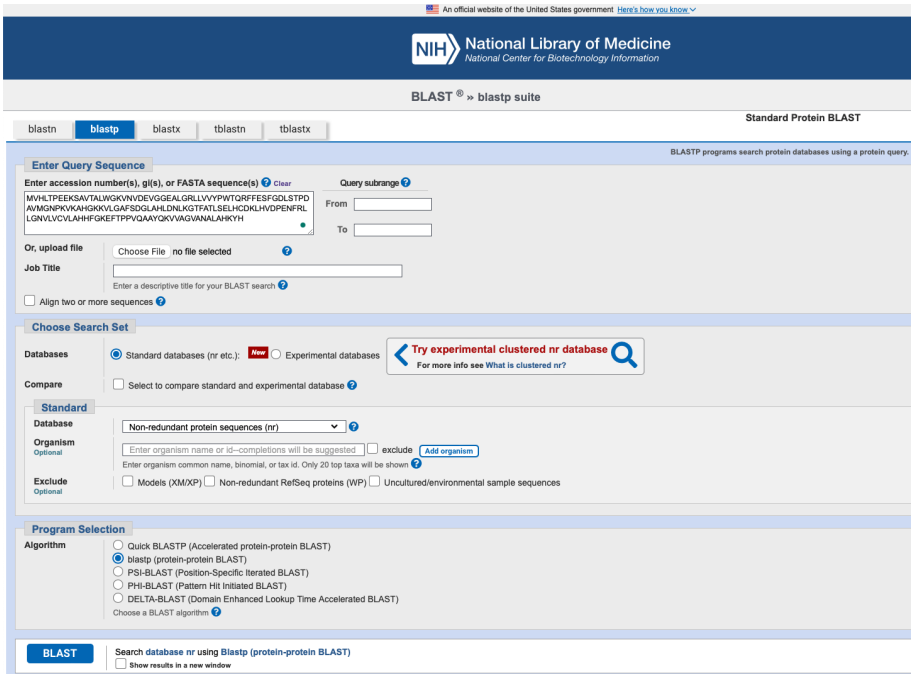

Figure 4.1. *HBB Homo sapiens blastp*

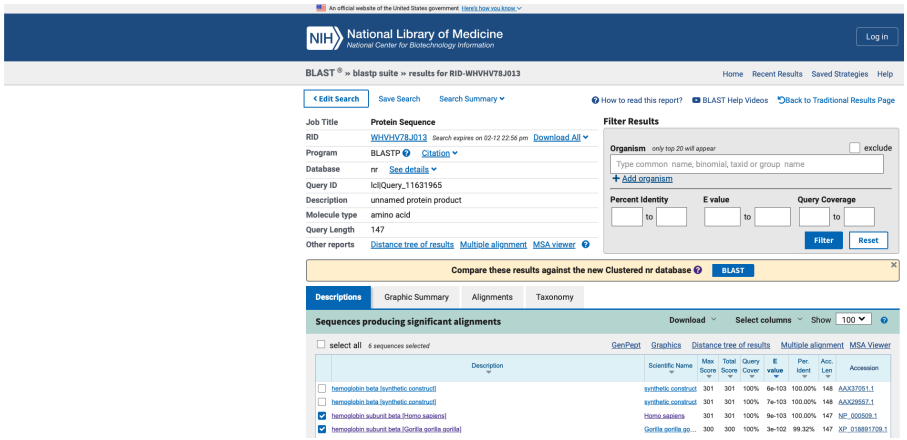

Figure 4.2. *blastp sequences selected I*

Based on the constructed phylogenetic tree the following we can make the following observations:

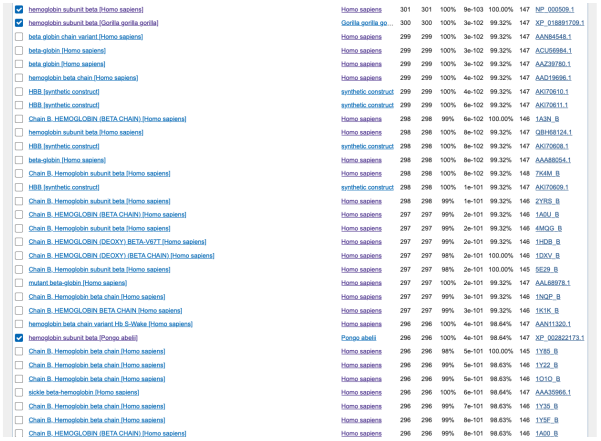

Figure 4.3. *blastp sequences selected II*

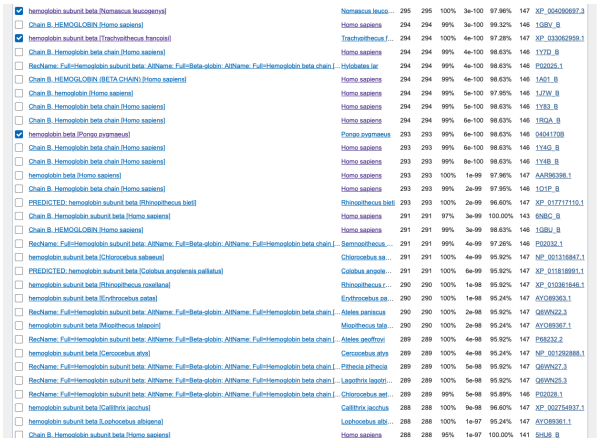

Figure 4.4. *blastp sequences selected III*

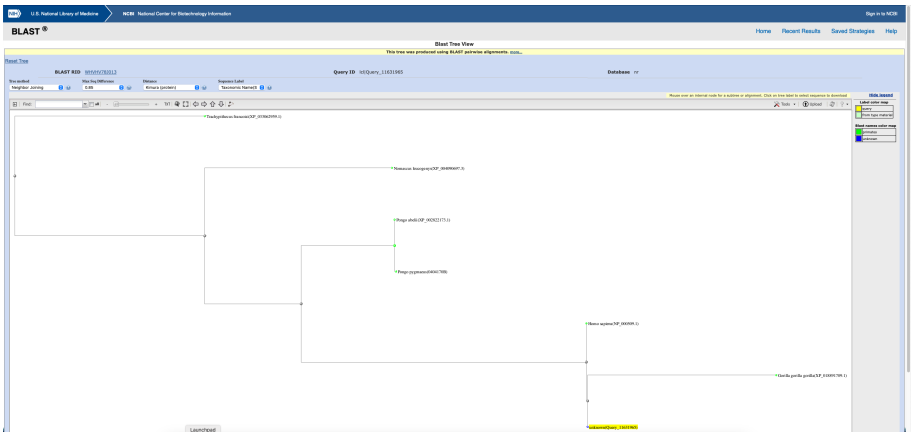

Figure 4.5. *NCBI Neighbor Joining Phylogenetic Tree*

- Homo sapiens and Gorilla gorilla gorilla sequences are closely related, indicating a close evolutionary relationship between these two species

- Both Pongo species (Pongo abelii and Pongo pygmaeus) branch off together in the phylogenetic tree, again indicating their close relationship with each other and their separation from the African apes (humans and gorillas).

- The Nomascus leucogenys (gibbon) and Trachypithecus francoisi (François' leaf monkey) are also placed on separate branches from the great apes, which is consistent with the evolutionary distance between these species.
- Both species of orangutan, Pongo abelii and Pongo pygmaeus, are grouped together, but they form a separate branch from the rest of the apes, indicating a divergence from the common ancestor they share with the African apes (gorillas, humans) and the Asian apes (Nomascus leucogenys and Trachypithecus francoisi).

4.2 MEGA Neighbor Joining Phylogenetic Tree

Having obtained all HBB protein sequences for all species, we import them to MEGA 4.7 and we perform multiple alignment 4.8. After alignment is successfully performed, we export aligned sequences in .meg file we open .meg file and we select the 'Construct/Test Neighbor-Joining tree with the appropriate settings 4.9. After clicking Ok we get the results 4.10, 4.11.

Based on the constructed phylogenetic tree the following we can make the following observations:

- Homo sapiens is shown to be most closely related to Gorilla gorilla gorilla, which is consistent with current understanding of hominid evolutionary relationships.
- The tree indicates that all the listed species have a common ancestor, with subsequent branching representing evolutionary divergence.
- The branch lengths represent the amount of genetic change, with the numbers presumably representing substitutions per site. Humans (Homo sapiens) and gorillas (Gorilla gorilla gorilla) have the shortest branches, suggesting fewer genetic changes since their divergence from a common ancestor compared to the other species.
- Both species of orangutan, Pongo abelii and Pongo pygmaeus, are grouped together, but they form a separate branch from the rest of the apes, indicating a divergence from the common ancestor they share with the African apes (gorillas, humans) and the Asian apes (Nomascus leucogenys and Trachypithecus francoisi).
- Nomascus leucogenys (gibbon) and Trachypithecus francoisi (François' leaf monkey) are grouped on the same branch, which is consistent with them both being Asian primates, but they diverged significantly from both the African apes and orangutans.

4.3 Differences between the two constructed phylogenetic trees

There are several differences between the two phylogenetic trees, both in terms of structure and the information they present. For brevity we will refer to the phylogenetic tree produced by MEGA11 as 'MEGA tree' and for phylogenetic tree produced by NCBI as 'NCBI tree'. Some of the differences are presented below:

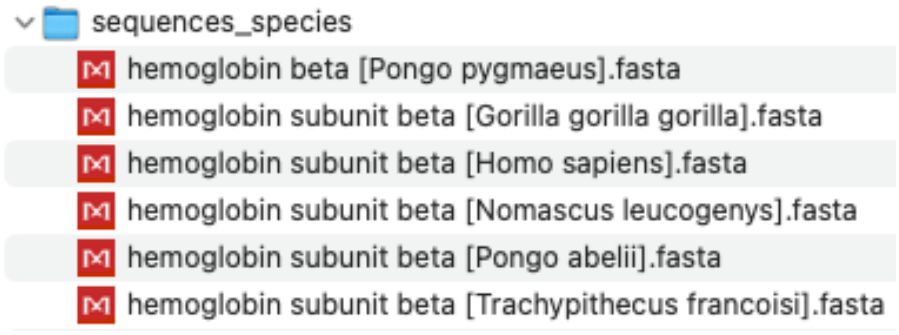

Figure 4.6. *Obtained sequences files in .fasta format*

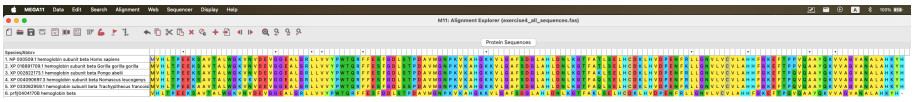

Figure 4.7. *All species sequences in .fasta format*

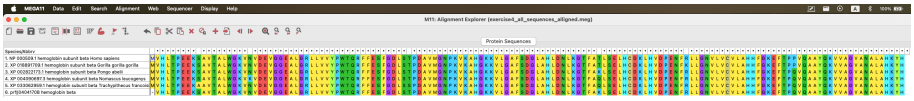

Figure 4.8. *All species aligned sequences*

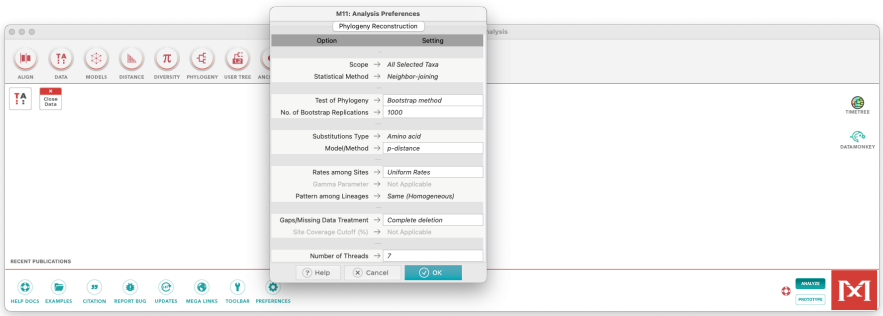

Figure 4.9. *Neighbor Joining Phylogenetic Tree settings*

- The MEGA tree has branch lengths with numerical values that represent genetic distances or the number of substitutions per site. This detail gives a sense of how much genetic change has occurred since the divergence from a common ancestor, while the tree, taken from the BLAST interface, does not show these numerical branch length values, making it harder to assess the exact amount of genetic change or distance.

- In MEGA tree, Homo sapiens is placed closer to Gorilla gorilla gorilla, while the two Pongo species (orangutans) are on a separate branch, and the gibbon (Nomascus leucogenys) and leaf monkey (Trachypithecus francoisi) are on another. In NCBI produced tree, the relationships seem to be consistent with the first in that humans are closer to gorillas, and the two orangutan species are grouped together. However, without the branch length data or bootstrap values, the strength of these relationships is not quantified.

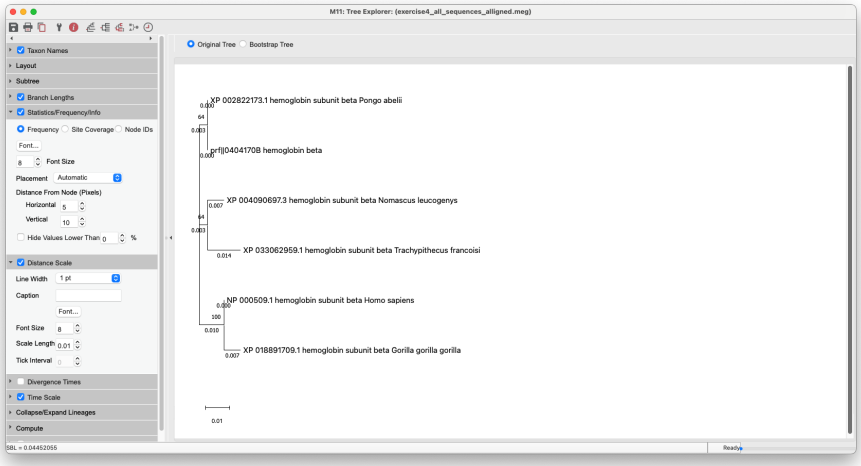

Figure 4.10. *Neighbor Joining Phylogenetic Tree*

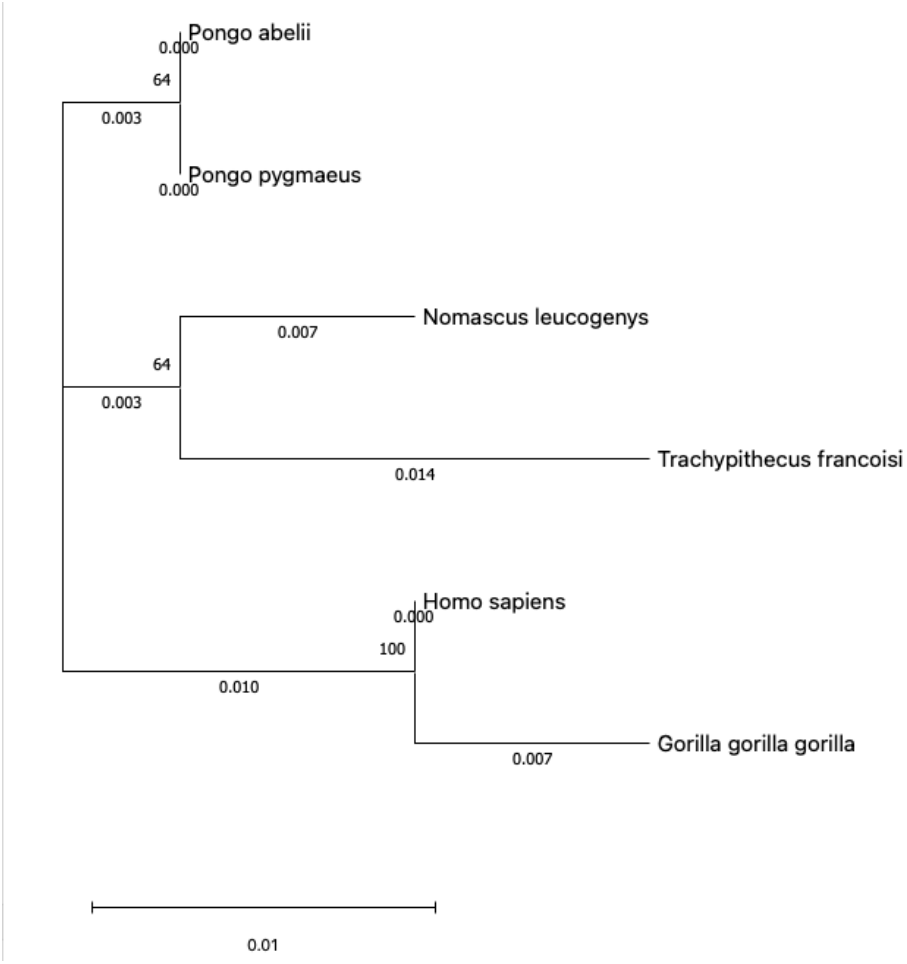

Figure 4.11. *Neighbor Joining Phylogenetic Tree clean*

- The MEGA tree includes a bootstrap value (64) at the node separating the orangutan species from the other primates, providing some measure of confidence in that grouping, while NCBI blastp produced tree does not have visible bootstrap values or other measures of statistical support.

- While both trees were constructed using the Neighbor-Joining method, the parameters such as the model of evolution (e.g., Kimura protein model in NCBI) and the exact settings differ, which affect the outcome of the tree.
- MEGA tree has a scale bar indicating the genetic distance, which is absent from the NCBI tree. This scale bar is crucial for interpreting the lengths of the branches in evolutionary terms.
- The MEGA tree is a more traditional representation, with clear delineation of branches and distances. The second tree is more schematic.

List of Abbreviations

BLAST	Basic Local Alignment Search Tool
blastp	Protein BLAST
bp	base pairs
EM	Electron Microscopy
NCBI	National Center for Biotechnology Information
NMR	Nuclear Magnetic Resonance
PCR	Polymerase Chain Reaction
PDB	Protein Data Bank