In [1]:

```
import os
os.chdir('/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/process_alignments')
```

In [1]:

```
import Bio
from Bio import SeqIO
from Bio import AlignIO
from Bio.Blast import NCBIWWW
from Bio.Blast import NCBIXML
from Bio import Entrez
from time import process_time
```

First step) Reading the sequence

In [3]:

```
seq = open('/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_
10seqs.fa').read()
print(seq)
```

>SUP35_Kla_AB039749
ATGTCAGACCAACAAAATCAAGACCAAGGGCAAGGCCAAGGTTACAATCAGTATAACCAATATGGCCA
GT
ACAACCAGTACTACAACCAACAGGGCTATCAAGGCTACAACGGCCAACAAGGTGCTCCTCAAGGCTAC
CA
AGCATATCAAGCTTATGGACAGCAGCCTCAAGGAGCCTACCAGGGCTACAACCCTCAACAAGCTCAAG
GC
TACCAACCTTACCAGGGCTACAATGCTCAGCAACAAGGTTACAACGCTCAGCAAGGCGGTCACAACAA
TA
ACTACAATAAAAATTATAATAATAAAAACAGTTACAATAACTATAATAAGCAGGGTTATCAAGGTGCT
CA
AGGATATAATGCACAACAGCCAACCGGTTACGCTGCTCCAGCACAGTCTTCATCCCAGGGTATGACTT
TG
AAAGATTTCCAAAACCAACAAGGCAGTACTAATGCAGCCAAGCCAAAGCCTAAGTTAAAGTTGGCCTC
TA
GCTCTGGTATTAAGTTAGTAGGTGCCAAGAAACCTGTAGCACCCAAAACTGAGAAAACTGATGAATCC
AA
GGAAGCAACTAAAACTACCGACGACAACGAAGAAGCACAATCTGAATTGCCCAAAATTGATGATTTGA
AA
ATCTCAGAGGCTGAAAAACCAAAAACTAAGGAGAATACCCCATCTGCTGATGATACTTCCTCAGAGAA
GA
CTACCAGCGCTAAGGCAGACACATCTACAGGAGGAGCGAACTCCGTGGATGCTCTAATCAAGGAACAA
GA
AGATGAGGTTGACGAAGAAGTCGTTAAAGATATGTTTGGTGGTAAGGATCATGTTTCCATCATTTTCA
TG
GGTCACGTCGATGCTGGTAAGTCAACAATGGGTGGTAACTTGTTATATCTGACTGGTTCTGTGGATAA
AA
GAACCGTTGAGAAATATGAGAGAGAGGCTAAAGAGGCTGGTAGACAAGGTTGGTATTTATCATGGGTG
AT
GGATACCAACAAAGAAGAAAGAAACGACGGTAAAACCATTGAAGTGGGTAGAGCGTACTTTGAAACTG
AA
AAGAGACGTTACACTATTTTGGATGCTCCCGGTCACAAAATGTATGTTTCTGAAATGATTGGTGGTGC
AT
CTCAAGCCGATATTGGTATTTTGGTTATTTCTGCTAGAAAGGGTGAATATGAAACTGGTTTCGAAAAG
GG
TGGTCAAACTCGTGAGCACGCTTTATTAGCCAAGACACAAGGTGTCAACAAAATGATTGTAGTTATCA
AC
AAGATGGATGATCCAACTGTGGGATGGGATAAGGAAAGATATGATCACTGTGTTGGTAACTTGACAAA
CT
TTTTGAAGGCTGTGGGTTACAATGTTAAGGAGGACGTCATTTTCATGCCAGTGTCTGGTTACACAGGT
GC
AGGTTTGAAGGAACGTGTCGATCCTAAGGACTGTCCATGGTATACTGGTCCATCTTTATTAGAATATC
TT
GACAATATGAAGACTACTGATCGTCATATCAATGCTCCATTCATGCTTCCAATTGCTTCTAAGATGAA
GG
ACATGGGTACTGTTGTGGAAGGTAAAATCGAATCTGGACACATTAGAAAGGGTAACCAAACTTTACTA
AT
GCCTAACAGGACCTCTGTTGAAATTCTGACCATTTATAACGAAACTGAAAGCGAAGTTGACATGGCTG
TT
TGTGGTGAGCAGGTTAGATTAAGAATTAAAGGTGTCGAAGAAGAAGAAATTTCTGCTGGTTTCGTTCT
AA
CCTCTCCAAAAAACCCAGTTAAGAATGTAACGAGATTTGTGGCTCAAATTGCTATTGTCGAATTGAAA
TC
GATCATGTCTGCTGGTTTCTCGTGCGTTATGCATATTCATACAGCTATCGAAGAAGTCACCGTCACAA
GA
TTGCTTCACAAGCTTGAGAAGGGGTCAAACAGAAAATCAAAGAAGCCTCCAGCATTTGCTAAAAAGGG
TA
TGAAGATCATTGCCGTTATCGAGACTAATGAACCGGTATGTGTTGAAACATACGATGATTACCCACAA
TT
GGGTAGATTCACTTTAAGAGATCAAGGTACCACCATTGCTATCGGTAAGATTGTGAAAATCCTTGAAA
AT

TGA
>SUP35_Agos_ATCC_10895_NM_211584
ATGTCGGAGGAAGATCAAATTCAATCGCAAGGCAACGACCAAGGCCAGTCGCAAGCCAAGGATCAAGG
TC
AAAATCAAGGTCAGGGGCAGCAAAATTTCGGCCAATACTACAACCCAAGTAACTTCCAGAATTACCAG
GG
GTATGTGCCTCAGGGAGGTTACCAGGCGTATGGCCAACAGGCAGGTGGGTACCAAGGCTATGCGCAAT
AC
AACCAGCAAGCCGGCGGGTACCAGGGTTATCAGGGATACCAGCAGTACAACCCAGCACAAGCTGGCTA
CC
AAGGTTACCAGCAATACAATGCGCAAGGTGGGTATCAGAGCTACAAGCAGTACAACTCACAGCCACAG
GG
GAACCGTAAGGGGAACCAGAGCTATGGTTACGGACAGGGCCAGTCAGCCACCGCTCCGGTGACGCTGA
AC
AACTTTGAGAAGGGAACTGTGCCGAATGCGACTGCTCCAAAACCAAAGAAGACCCTCAAGTTGGCTTC
CA
GCTCCGGTATCAAGCTAGTTGGTGCTAAGAAACCTGTCGCGAAGAAAGAAGAGGCTAAGGCGGAGGAG
CC
AACAAAGGAGGAGCCTAAGAGCAGCGCAGAGGGCGCTCCTAAGAGTGAGGATGCTACCGCTTCCTCTG
AA
GATAAGGCAGTCCCAAGTATTGAGAAGCTAAGTATTTCGGAAGCCGATACCGCGAAGAAAGACACAGC
GG
ACGCGGCCGGTGCCACCTCATCGGATGCTCTGATCAAGGAACAGGAAGACGAAGTTGATGAAGAAGTT
GT
GAAGGACATGTTTGGTGGAAAGGACCATGTTTCTATCATCTTTATGGGTCACGTCGATGCGGGGAAAT
CC
ACCATGGGCGGAAACTTGTTGTATCTAACCGGCTCGGTAGACAAAAGAACAGTTGAGAAGTATGAGAA
GG
AGGCGAAGGAAGCAGGGAGACAGGGTTGGTACTTGTCTTGGATTATGGATACAAATAAGGAGGAGAGA
AA
CGATGGTAAGACCATTGAAGTGGGTAGATCCTACTTTGAAACTGAAAAGAGACGGTATACCATTTTGG
AT
GCACCAGGCCACAAAATGTACGTCTCTGAAATGATTGGTGGTGCGTCTCAAGCCGATGTTGGTATATT
GG
TCATCTCAGCCAGAAAGGGTGAGTACGAGACTGGTTTTGAGAGAGGTGGACAAACGCGTGAGCATGCG
CT
ATTAGCGAAGACCCAGGGTGTCAATAAAATGGTGGTCGTCGTGAACAAAATGGATGACCCCACGGTAA
AC
TGGGACAAGGCACGTTATGACCAGTGTATTAAGAACGTCTCCAATTTCTTGCAAGCTATTGGCTACAA
CG
TTAAGGAGGATGTTATGTATATGCCTGTTTCTGGGTTTACTGGTGCTGGTCTGAAGGACCGCGTTGAC
AA
GAAGGACTGTCCTTGGTACGATGGTCCTTCCCTTTTGGAATACCTAGACAACATGCAGCATGTTGACC
GC
TTCATCAACGCGCCTTTCATGCTTCCAATTGCAAGCAAGATGAAGGATATGGGTACAGTGGTCGAAGG
TA
AAATCGAATCTGGTCATATTAAGAAGGGTAATCAAACACTACTGATGCCAAACAAAATTCCAGTGGAG
AT
TCTAGCCATCCAAAATGAGACCGAACAGGAAGTTGATATGGCTGTGTGTGGTGAGCAGGTCAGATTGA
GA
CTGAAGGGAGTTGAGGAGGAAGACATTTCTGCAGGTTTTGTTTTGACTTCGCCAAAGAACCCTGTCAA
GA
ACGTCACGAAGTTTGTTGCCCAGATTGCCATTGTGGAATTGAAGTCCATTATGTCAGCAGGTTTCTCC
TG
TGTGATGCACGTGCATACCGCTATTGAGGAAGTCTCTATCACTAGACTACTTCATAAGCTGGAGAAGG
GC
ACCAACAGAAAGTCCAAGAAGCCACCTGCCTTTGCTAAGAAGGGCATGAAGATAATAGCGGTCCTTGA
GA
CAGAGGAGCCAGTATGTGTGGAAACCTACCAGGATTACCCACACTTGGGTAGATTCACATTGAGAGAC
CA
GGGTATTACAATCGCCATCGGTAAGATTGTGAAGATCTTGGAATGA

>SUP35_Scer_74-D694_GCA_001578265.1
ATGTCGGATTCAAACCAAGGCAACAATCAGCAAAACTACCAGCAATACAGCCAGAACGGTAACCAACA
AC
AAGGTAACAACAGATACCAAGGTTATCAAGCTTACAATGCTCAAGCCCAACCTGCAGGTGGGTACTAC
CA
AAATTACCAAGGTTATTCTGGGTACCAACAAGGTGGCTATCAACAGTACAATCCCGACGCCGGTTACC
AG
CAACAGTATAATCCTCAAGGAGGCTATCAACAGTACAATCCTCAAGGCGGTTATCAGCAGCAATTCAA
TC
CACAAGGTGGCCGTGGAAATTACAAAAACTTCAACTACAATAACAATTTGCAAGGATATCAAGCTGGT
TT
CCAACCACAGTCTCAAGGTATGTCTTTGAACGACTTTCAAAAGCAACAAAAGCAGGCCGCTCCCAAAC
CA
AAGAAGACTTTGAAGCTTGTCTCCAGTTCCGGTATCAAGTTGGCCAATGCTACCAAGAAGGTTGGCAC
AA
AACCTGCCGAATCTGATAAGAAAGAGGAAGAGAAGTCTGCTGAAACCAAAGAACCAACTAAAGAGCCA
AC
AAAGGTCGAAGAACCAGTTAAAAAGGAGGAGAAACCAGTCCAGACTGAAGAAAAGACGGAGGAAAAAT
CG
GAACTTCCAAAGGTAGAAGACCTTAAAATCTCTGAATCAACACATAATACCAACAATGCCAATGTTAC
CA
GTGCTGATGCCTTGATCAAGGAACAGGAAGAAGAAGTGGATGACGAAGTTGTTAACGATATGTTTGGT
GG
TAAAGATCACGTTTCTTTAATTTTCATGGGTCATGTTGATGCCGGTAAATCTACTATGGGTGGTAATC
TA
CTATACTTGACTGGCTCTGTGGATAAGAGAACTATTGAGAAATATGAAAGAGAAGCCAAGGATGCAGG
CA
GACAAGGTTGGTACTTGTCATGGGTCATGGATACCAACAAAGAAGAAAGAAATGATGGTAAGACTATC
GA
AGTTGGTAAGGCCTACTTTGAAACTGAAAAAAGGCGTTATACCATATTGGATGCTCCTGGTCATAAAA
TG
TACGTTTCCGAGATGATCGGTGGTGCTTCTCAAGCTGATGTTGGTGTTTTGGTCATTTCCGCCAGAAA
GG
GTGAGTACGAAACCGGTTTTGAGAGAGGTGGTCAAACTCGTGAACACGCCCTATTGGCCAAGACCCAA
GG
TGTTAATAAGATGGTTGTCGTCGTAAATAAGATGGATGACCCAACCGTTAACTGGTCTAAGGAACGTT
AC
GACCAATGTGTGAGTAATGTCAGCAATTTCTTGAGAGCAATTGGTTACAACATTAAGACAGACGTTGT
AT
TTATGCCAGTATCCGGCTACAGTGGTGCAAATTTGAAAGATCACGTAGATCCAAAAGAATGCCCATGG
TA
CACCGGCCCAACTCTGTTAGAATATCTGGATACAATGAACCACGTCGACCGTCACATCAATGCTCCAT
TC
ATGTTGCCTATTGCCGCTAAGATGAAGGATCTAGGTACCATCGTTGAAGGTAAAATTGAATCCGGTCA
TA
TCAAAAAGGGTCAATCCACCCTACTGATGCCTAACAAAACCGCTGTGGAAATTCAAAATATTTACAAC
GA
AACTGAAAATGAAGTTGATATGGCTATGTGTGGTGAGCAAGTTAAACTAAGAATCAAAGGTGTTGAAG
AA
GAAGACATTTCACCAGGTTTTGTACTAACATCGCCAAAGAACCCTATCAAGAGTGTTACCAAGTTTGT
AG
CTCAAATTGCTATTGTAGAATTAAAATCTATCATAGCAGCCGGTTTTTCATGTGTTATGCATGTTCAT
AC
AGCAATTGAAGAGGTACATATTGTTAAGTTATTGCACAAATTAGAAAAGGGTACCAACCGTAAGTCAA
AG
AAACCACCTGCTTTTGCTAAGAAGGGTATGAAGGTCATCGCTGTTTTAGAAACTGAAGCTCCAGTTTG
TG
TGGAAACTTACCAAGATTACCCTCAATTAGGTAGATTCACTTTGAGAGATCAAGGTACCACAATAGCA
AT
TGGTAAAATTGTTAAAATTGCCGAGTAA
>SUP35_Spar_A12_Liti

ATGTCGGATTCAAACCAAGGTAACAATCAGCAAAGCTACCAGCAATACGGCCAAAACCCTAACCAACA
AC
AAGGTAACAACAGATACCAAGGTTATCAAGCTTACAATGCTCAATCCCAACCTGCAGGTGGGTATTAC
CA
AAATTACCAAGGTTATTCTGGATACCAACAAGGCAGCTACCAACAGTACAACCCAGATGCCGGTTACC
AG
CAACAATATAATCCCCAAGGTGGCTATCAACAGTACAATCCTCAAGGCGGTTATCAGCAACAGTTCAA
TC
CACAAGGTGGCCGTGGCAATTACAAAAACTTCAACTACAATAACAATGCACAAGGATATCAAGCTGGT
TT
CCAACCACAGTCTCAAGGTATGTCTTTGAACGACTTCCAAAAGCAACAAAAGCAAGCCGCTCCCAAGC
CA
AAGAAAACTTTGAAGCTTGTTTCCAGTTCTGGTATCAAGTTGGCCAATGCTACCAAGAAGGTCGACAC
AA
AACCTGCTGAATCTGAAAAGAAGAAGGAAGACAAACCCACCGAAAACAAAGAACCAACAAAACTCGAA
GA
ACCAGTTAAAAAAGAGGAGAAACTAGTCAAGACCGAAGAAAAAAAGGAGGAGAAATCGGAACTCCCAA
AG
GTAGAAGACCTGAAAATCTCTGAATCAACAGATAACACCAACAATGCCAATGTTAACAGTGCTGATGC
CT
TGATCAAAGAACAGGAAGAAGAAGTCGATGACGAAGTTGTTAACGACATGTTCGGGGGTAAAGATCAC
GT
TTCCTTAATTTTCATGGGTCATGTTGATGCTGGTAAATCTACTATGGGTGGTAATCTACTGTACTTGA
CT
GGCTCTGTCGATAAGAGAACCATTGAGAAATATGAAAGAGAGGCCAAGGATGCGGGAAGACAGGGTTG
GT
ACTTGTCATGGGTCATGGATACCAACAAAGAAGAAAGAAATGATGGTAAAACTATCGAAGTTGGTAAA
GC
CTACTTCGAAACTGAAAAAAGGCGTTATACCATATTGGATGCCCCTGGTCATAAAATGTACGTTTCCG
AG
ATGATCGGTGGTGCTTCTCAAGCTGATGTTGGTGTTTTGGTCATTTCCGCCAGAAAGGGTGAGTACGA
AA
CTGGTTTTGAAAGAGGTGGTCAAACACGTGAACACGCTCTATTAGCCAAGACCCAAGGTGTTAATAAG
AT
GGTTGTTGTCGTAAATAAGATGGATGACCCAACTGTTAACTGGTCTAAGGAACGTTATGACCAATGTG
TG
AGTAATGTCAGCAATTTCTTGAGAGCAATTGGCTACAATATCAAGACGGATGTTGTATTTATGCCAGT
AT
CTGGTTACAGTGGTGCAAACCTGAAAGATCACGTAGACCCAAAAGAATGCCCATGGTACACTGGCCCA
AC
TCTGTTGGAATATTTGGACACAATGAACCACGTCGACCGTCACATCAATGCTCCTTTCATGTTGCCTA
TT
GCCGCTAAAATGAAGGATTTAGGTACCATCGTTGAAGGTAAGATTGAATCTGGTCATATCAAAAAGGG
CC
AATCCACGCTATTGATGCCTAACAAAACTGCAGTGGAAATTCAAAACATTTACAACGAAACAGAAAAT
GA
AGTTGACATGGCTATGTGTGGTGAACAAGTTAAACTAAGAATCAAAGGTGTTGAAGAAGAAGACATTT
CA
CCAGGGTTCGTACTAACATCACCAAAGAACCCTATCAAAAGTGTTACCAAGTTTGTAGCTCAAATTGC
AA
TTGTCGAATTAAAATCTATTATTGCTGCTGGGTTTTCATGCGTTATGCATGTTCATACAGCAATTGAA
GA
AGTTCATATTGTTAAATTATTGCACAAATTGGAAAAGGGTACTAACCGTAAATCAAAGAAACCACCTG
CG
TTTGCTAAGAAAGGTATGAAGGTCATTGCTGTCTTAGAGACTGAAGCTCCAGTTTGTGTTGAAACTTA
CC
AAGATTACCCTCAATTAGGTAGATTCACTTTGAGAGATCAAGGTACCACAATAGCAATTGGTAAGATT
GT
TAAAATTGCTGAATAA
>SUP35_Smik_IFO1815T_30
ATGTCTGATTCAAACCAAGGTAATAATCAGCAAAACTACCAGCAATACAACCAAAACCCCAACCAACA

GC
AAGGTAACAATAGATACCAAGGTTATCAAGCTTATAATGCTCAAGCGCAACCTGCAAGTGGCCATTAC
CA
AAACTACCAAGGCTACTCTGGGTACCAACAAGGCGCTTTTCAACAGTACAACCCAGAGGCTGGCTACC
AA
CAACAGTACAACCCCCAAGGTGGCTATCAACAATACAATTCCCAGGGCGGTTATCAGCAACAATTCAA
TC
CACAAGGCGGTCGTGGAAATTACCAAAACATCAATTATAACAGCAATATACAAGGATACCAAGCTGGT
TT
TCAACCACAGTCTCAAAGTATGTCTTTAAATGACTTCCAAAAGCAACAAAAGCAAGCTGCGCCTAAGC
CA
AAGAAGACTTTGAAGCTTGTTTCCAGTTCTGGTATCAAGCTAGCTAATGCTACCAAGAAAGTTGATAC
AA
AACCTTCCGAAACTGAAAAGAAAGAGGAAGTCAAGCCTGTCGAAACCAAAAAACCAATTAAAGTTGAA
GA
ACCAGTCAAGAAAGAGGAAATGCCAGTAAAGGCCGAAGAAAAGAAGGAAGAAAAGTCTGAACTGCCAA
AA
GTAGGAGACTTAAAAATTTCTGAATCGACAGATAACAGCAACACTACCAGTGCTACCAGTGCTGATGC
CT
TGATCAAAGAACAGGAAGAAGAAGTTGATGATGAGGTTGTTAATGATATGTTCGGGGGCAAAGATCAC
GT
TTCTTTAATCTTTATGGGTCATGTCGATGCTGGTAAGTCTACTATGGGTGGTAACTTACTGTACTTGA
CA
GGCTCTGTGGATAAGAGAACTATAGAGAAATATGAAAGAGAGGCCAAGGATGCTGGTAGACAAGGTTG
GT
ATTTGTCATGGGTCATGGATACCAACAAAGAAGAAAGAAATGATGGTAAGACCATTGAAGTTGGTAAG
GC
CTACTTCGAAACTGAGAAAAGGCGTTATACTATATTGGATGCTCCTGGTCATAAAATGTACGTTTCCG
AA
ATGATTGGTGGTGCTTCTCAAGCTGATGTCGGTGTTTTAGTCATTTCTGCCAGAAAGGGTGAATACGA
GA
CTGGTTTTGAAAGAGGTGGTCAAACTCGTGAACATGCCTTATTGGCCAAGACCCAAGGTGTTAATAAG
AT
GGTTGTCGTCGTAAATAAAATGGATGATCCAACTGTCAATTGGTCCAAGGAACGTTATGATCAATGTG
TA
AGTAATGTCAGCAATTTCTTGAAAGCCATTGGCTATAACATCAAGACGGATGTTGTGTTTATGCCAGT
AT
CAGGTTACAGTGGTGCAAACTTGAAAAACCGTGTAGATCCAAAAGAATGTCCATGGTATACTGGCCCA
AC
TCTATTGGAGTATTTAGATACAATGAACCATGTCGATCGTCACATCAACGCTCCATTCATGTTGCCTA
TT
GCCGCTAAGATGAAGGATCTGGGCACGATCGTTGAAGGTAAGATTGAATCTGGTCATATCAAGAAGGG
TC
AATCCACCCTATTAATGCCCAACAAAACGGCAGTGGAAATTCAAAACATTTATAATGAAACTGAAAAC
GA
AGTTGACATGGCTATGTGTGGTGAGCAAGTCAAGCTGAGAATTAAGGGTGTTGAAGAAGAAGATATTT
CC
CCAGGGTTTGTACTAACATCACCAAAGAATCCTATCAAGAGTGTCACCAAGTTTGTGGCCCAAATTGC
CA
TTGTCGAACTAAAATCTATTATAGCTGCTGGGTTTTCATGTGTTATGCATGTTCATACAGCAATTGAA
GA
AGTTCATATTGTTAAATTATTGCACAAATTGGAAAAGGGTACTAACCGTAAATCAAAGAAGCCACCTG
CT
TTTGCCAAGAAGGGTATGAAAGTCATTGCTGTTTTAGAGACTGAAGCTCCGGTTTGTGTCGAAACTTA
CC
AGGATTACCCTCAGTTAGGTAGATTCACCTTAAGAGATCAAGGTGCTACAATAGCAATTGGTAAGATC
GT
TAAGATTGCTGAATGA
>SUP35_Skud_IFO1802T_36
ATGTCAGATCCAAATCAAGGTAACAATCAACAACAATACGGTCAAAATCCTAACCAACAGCAAGGCAA
TA

ACAAGTACCAAGGTTATCAAGCTTACAATGCTCAAGCCCAACCTGCAAACGGCTACTATCAAAACTAC
CA
AGGATTTGCCGGCTATCAACAAGGCGGCTACCAACAGTATGACCGAGACGCCGGTAGTCAGCAACAAT
AC
AACCCACAAGGTGGCTATCAACAGTACCCCCCACAGGAAGGTTACCAACAACAGTCCAATCCACAAGG
TG
GTCGTGGTAATTACAAAAGCTTCAACTATAACAACAATGTTCAAGGGTATCAAGCTGGCTTCCAACCG
CA
ATCTCAAGGTATGTCTCTGAACGACTTCCAAAAGCAACAAAAGCAAACTGCTCCTAAGCCAAAGAAGA
CT
TTGAAGCTTGTTTCAAGCTCTGGTATCAAGTTAGCTAATGCTACGAAGAAGGTCAGCACAAAGCCTGC
CG
AAACCGAAAAGAAAGAGGAAGACAAGCCTGTCAAAACCGAGGAACAAATCAAGGTCGAAGAACCAATT
AA
AGAAGAGAAAAAAGCAGTCAAGACTGAGGAAAAGAAAGAGGTAGAATCTGAACTACCAAGGGTCGAAG
AC
TTGAAGATATCCGAATCGACTGATAATAGCAATACTGCTAACGTCACAAGTGCAGACACTTTGATAAA
AG
AACAAGAAGAAGAAGTTGATGACGAAGTTGTTAATGATATGTTTGGCGGTAAAGATCACGTTTCTTTG
AT
TTTCATGGGACATGTTGATGCTGGTAAGTCTACTATGGGTGGTAATCTGCTATACTTGACTGGCTCCG
TG
GATAAAAGAACAATTGAAAAATACGAAAGAGAAGCTAAGGATGCTGGTAGGCAAGGTTGGTACTTGTC
AT
GGGTCATGGATACAAATAAGGAAGAGAGAAACGACGGTAAGACCATTGAAGTTGGTAAAGCTTATTTC
GA
AACTGAGAAAAGACGTTATACCATTCTGGATGCTCCTGGTCATAAAATGTACGTTTCCGAAATGATTG
GT
GGTGCTTCGCAAGCCGATGTTGGTGTTTTAGTCATATCTGCCAGAAAGGGTGAATATGAAACCGGTTT
CG
AAAGAGGTGGCCAAACACGTGAACACGCCCTATTGGCCAAGACCCAGGGTGTTAATAAAATGGTTGTT
GT
CGTAAATAAGATGGATGATGCAACCGTCAACTGGTCCAAAGAACGTTACGATCAATGTGTAGGTAATG
TT
AGCAACTTTTTGAAAGCGATTGGTTACAACATCAAGACAGATGTTGCGTTTATGCCAGTATCAGGTTA
CA
GCGGTGCAAACTTGAAAAATCGTGTAGATCCAAAGGAGTGTTCATGGTATACCGGCCCAACTCTGTTG
GA
ATACCTGGACACAATGAACCACGTCGATCGTCATGTCAATGCTCCATTCATGTTGCCTATTGCTGCTA
AG
ATGAAGGATTTAGGTACCATAGTTGAAGGCAAGATCGAATCCGGTCATATCAAGAAGGGTCAATCCAC
CT
TGTTGATGCCTAACAAAACCTCAGTGGAAATTCAAAACATTTACAACGAAACTGAAAACGAAGTTGAC
AT
GGCTATGTGCGGTGAACAGGTTAAGTTGAGAATTAAAGGAGTTGAAGAAGAAGACATCTCTCCAGGGT
TT
GTGTTAACATCACCAAAGAACCCTATCAAAAGTGTTACCAAATTTGTTGCCCAAATTGCCATTGTAGA
GT
TAAAGTCCATTATAGCTGCTGGGTTTTCATGTGTCATGCACGTTCATACAGCCATTGAAGAGGTTCAT
AT
TGTTAAATTATTGCACAAATTAGAAAAGGGTAGCAATCGTAAATCGAAGAAGCCACCTGCTTTCGCTA
AG
AAAGGTATGAAGGTTATTGCTGTTTTAGAGACTGAAGCTCCAGTTTGTGTCGAAGCTTACCAAGATTA
TC
CTCAATTGGGTAGATTCACTCTAAGAGATCAAGGTACCACAATCGCCATCGGTAAAATTGTTAAGATT
GC
TGAATAA
>SUP35_Sbou_unique28_CM003560
ATGTCGGATTCAAACCAAGGCAACAATCAGCAAAACTACCAGCAATACAGCCAGAACGGTAACCAACA
AC
AAGGTAACAACAGATACCAAGGTTATCAAGCTTACAATGCTCAAGCCCAACCTGCAGGTGGGTACTAC

CA
AAATTACCAAGGTTATTCTGGGTACCAACAAGGTGGCTATCAACAGTACAATCCCGACGCCGGTTACC
AG
CAACAGTATAATCCTCAAGGAGGCTATCAACAGTACAATCCTCAAGGCGGTTATCAGCAGCAATTCAA
TC
CACAAGGTGGCCGTGGAAATTACAAAAACTTCAACTACAATAACAATTTGCAAGGATATCAAGCTGGT
TT
CCAACCACAGTCTCAAGGTATGTCTTTGAACGACTTTCAAAAGCAACAAAAGCAGGCCGCTCCCAAAC
CA
AAGAAGACTTTGAAGCTTGTCTCCAGTTCCGGTATCAAGTTGGCCAATGCTACCAAGAAGGTTGACAC
AA
AACCTGCCGAATCTGATAAGAAAGAGGAAGAGAAGTCTGCTGAAACCAAAGAACCAACTAAAGAGCCA
AC
AAAGGTCGAAGAACCAGTTAAAAAGGAGGAGAAACCAGTCCAGACTGAAGAAAAGACGGAGGAAAAAT
CG
GAACTTCCAAAGGTAGAAGACCTTAAAATCTCTGAATCAACACATAATACCAACAATGCCAATGTTAC
CA
GTGCTGATGCCTTGATCAAGGAACAGGAAGAAGAAGTGGATGACGAAGTTGTTAACGATATGTTTGGT
GG
TAAAGATCACGTTTCTTTAATTTTCATGGGTCATGTTGATGCCGGTAAATCTACTATGGGTGGTAATC
TA
CTATACTTGACTGGCTCTGTGGATAAGAGAACTATTGAGAAATATGAAAGAGAAGCCAAGGATGCAGG
CA
GACAAGGTTGGTACTTGTCATGGGTCATGGATACCAACAAAGAAGAAAGAAATGATGGTAAGACTATC
GA
AGTTGGTAAGGCCTACTTTGAAACTGAAAAAAGGCGTTATACCATATTGGATGCTCCTGGTCATAAAA
TG
TACGTTTCCGAGATGATCGGTGGTGCTTCTCAAGCTGATGTTGGTGTTTTGGTCATTTCCGCCAGAAA
GG
GTGAGTACGAAACCGGTTTTGAGAGAGGTGGTCAAACTCGTGAACACGCCCTATTGGCCAAGACCCAA
GG
TGTTAATAAGATGGTTGTCGTCGTAAATAAGATGGATGACCCAACCGTTAACTGGTCTAAGGAACGTT
AC
GACCAATGTGTGAGTAATGTCAGCAATTTCTTGAGAGCAATTGGTTACAACATTAAGACAGACGTTGT
AT
TTATGCCAGTATCCGGCTACAGTGGTGCAAATTTGAAAGATCACGTAGATCCAAAAGAATGCCCATGG
TA
CACCGGCCCAACTCTGTTAGAATATCTGGATACAATGAACCACGTCGACCGTCACATCAATGCTCCAT
TC
ATGTTGCCTATTGCCGCTAAGATGAAGGATCTAGGTACCATCGTTGAAGGTAAAATTGAATCCGGTCA
TA
TCAAAAAGGGTCAATCCACCCTACTGATGCCTAACAAAACCGCTGTGGAAATTCAAAATATTTACAAC
GA
AACTGAAAATGAAGTTGATATGGCTATGTGTGGTGAGCAAGTTAAACTAAGAATCAAAGGTGTTGAAG
AA
GAAGACATTTCACCAGGTTTTGTACTAACATCGCCAAAGAACCCTATCAAGAGTGTTACCAAGTTTGT
AG
CTCAAATTGCTATTGTAGAATTAAAATCTATCATAGCAGCCGGTTTTTCATGTGTTATGCATGTTCAT
AC
AGCAATTGAAGAGGTACATATTGTTAAGTTATTGCACAAATTAGAAAAGGGTACCAACCGTAAGTCAA
AG
AAACCACCTGCTTTTGCTAAGAAGGGTATGAAGGTCATCGCTGTTTTAGAAACTGAAGCTCCAGTTTG
TG
TGGAAACTTACCAAGATTACCCTCAATTAGGTAGATTCACTTTGAGAGATCAAGGTACCACAATAGCA
AT
TGGTAAAATTGTTAAAATCGCCGAGTAA
>SUP35_Scer_beer078_CM005938
ATGTCGGATTCAAACCAAGGCAACAATCAGCAAAACTA
CCAGCAATACAGCCAGAACGGTAACCAACAACAAGGTAACAACAGATACCAAGGTTATCAAGCTTACA
AT
GCTCAAGCCCAACCTGCAGGTGGGTACTACCAAAATTACCAAGGTTATTCTGGGTACCAACAAGGTGG

CT
ATCAACAGTACAATCCTCAAGGCGGTTATCAGCAGCAATTCAATCCACAAGGTGGCCGTGGAAATTAC
AA
AAACTTCAACTACAATAACAATTTGCAAGGATATCAAGCTGGTTTCCAACCACAGTCTCAAGGTATGT
CT
TTGAACGACTTTCAAAAGCAACAAAAGCAGGCCGCTCCCAAACCAAAGAAGACTTTGAAGCTTGTCTC
CA
GTTCCGGTATCAAGTTGGCCAATGCTACCAAGAAGGTTGACACAAAACCTGCCGAATCTGATAAGAAA
GA
GGAAGAGAAGTCTGCTGAAACCAAAGAACCAACTAAAGAGCCAACAAAGGTCGAAGAACCAGTTAAAA
AG
GAGGAGAAACCAGTCCAGACTGAAGAAAAGACGGAGGAAAAATCGGAACTTCCAAAGGTAGAAGACCT
TA
AAATCTCTGAATCAACACATAATACCAACAATGCCAATGTTACCAGTGCTGATGCCTTGATCAAGGAA
CA
GGAAGAAGAAGTGGATGACGAAGTTGTTAACGATATGTTTGGTGGTAAAGATCACGTTTCTTTAATTT
TC
ATGGGTCATGTTGATGCCGGTAAATCTACTATGGGTGGTAATCTACTATACTTGACTGGCTCTGTGGA
TA
AGAGAACTATTGAGAAATATGAAAGAGAAGCCAAGGATGCAGGCAGACAAGGTTGGTACTTGTCATGG
GT
CATGGATACCAACAAAGAAGAAAGAAATGATGGTAAGACTATCGAAGTTGGTAAGGCCTACTTTGAAA
CT
GAAAAAAGGCGTTATACCATATTGGATGCTCCTGGTCATAAAATGTACGTTTCCGAGATGATCGGTGG
TG
CTTCTCAAGCTGATGTTGGTGTTTTGGTCATTTCCGCCAGAAAGGGTGAGTACGAAACCGGTTTTGAG
AG
AGGTGGTCAAACTCGTGAACACGCCCTATTGGCCAAGACCCAAGGTGTTAATAAGATGGTTGTCGTCG
TA
AATAAGATGGATGACCCAACCGTTAACTGGTCTAAGGAACGTTACGACCAATGTGTGAGTAATGTCAG
CA
ATTTCTTGAGAGCAATTGGTTACAACATTAAGACAGACGTTGTATTTATGCCAGTATCCGGCTACAGT
GG
TGCAAATTTGAAAGATCACGTAGATCCAAAAGAATGCCCATGGTACACCGGCCCAACTCTGTTAGAAT
AT
CTGGATACAATGAACCACGTCGACCGTCACATCAATGCTCCATTCATGTTGCCTATTGCCGCTAAGAT
GA
AGGATCTAGGTACCATCGTTGAAGGTAAAATTGAATCCGGTCATATCAAAAAGGGTCAATCCACCCTA
CT
GATGCCTAACAAAACCGCTGTGGAAATTCAAAATATTTACAACGAAACTGAAAATGAAGTTGATATGG
CT
ATGTGTGGTGAGCAAGTTAAACTAAGAATCAAAGGTGTTGAAGAAGAAGACATTTCACCAGGTTTTGT
AC
TAACATCGCCAAAGAACCCTATCAAGAGTGTTACCAAGTTTGTAGCTCAAATTGCTATTGTAGAATTA
AA
ATCTATCATAGCAGCCGGTTTTTCATGTGTTATGCATGTTCATACAGCAATTGAAGAGGTACATATTG
TT
AAGTTATTGCACAAATTAGAAAAGGGTACCAACCGTAAGTCAAAGAAACCACCTGCTTTTGCTAAGAA
GG
GTATGAAGGTCATCGCTGTTTTAGAAACTGAAGCTCCAGTTTGTGTGGAAACTTACCAAGATTACCCT
CA
ATTAGGTAGATTCACTTTGAGAGATCAAGGTACCACAATAGCAATTGGTAAAATTGTTAAAATTGCCG
AG
TAA
>SUP35_Sarb_H-6_chrXIII_CM001575
ATGTCTGATCCAACTAATGGTAATAATGAGCAGAGCTCTCAACAGCAAGGCCAAAACCCTAGCCAACA
GC
AAGGTAACAACAGATATCAAGGCTATCAAGCTTATAATGCTCAAACTCAACCAGCAGGTGGCTATTAC
CA
AAACTACCAAGGCTATGCTGGCTACCAACAAGGTGGATACCAACAGTTCACTCCAGAGGCTGGTTACC
AA

CAACAGTACAACCCCCAAGGTGGCTATCAACAGTTCAACCCCCAAGGCGGTTATCAACAACAGTTCAA
TC
CACAAGGTGGCCGTGGCAACTACAAAAACTTTAACAATAACAACCAACAAGGATACCAAAGTAATTTT
CA
ACCGCAATCTCAAGGTATGTCTTTAAACGATTTCCAAAAGCAACAAAAGCAATCCACTCCTAAACCGA
AG
AAAACTTTGAAGCTTGTTTCAAGTTCCGGTATCAAGTTGGCTAATGCTACTAAGAAGGTCGATACAAA
GC
CCGTCGAAACCGAAAAGAAAGAAGAAGACAAGCCTGTTGAAACTAAAAAACCAACGAAGGTCGAAGAA
CC
AGCTAAGAAAGAGGAAGAGCCCGTCAAGGCTGAAGAAGCAAAGGAGGAAAAATCAGAACTACCAAAGG
TT
GAAGACTTAAAAATATCTGAACCAACCGATAACAGCAACGCTGCTAGCGTCAACAATGCAGACGCCTT
AA
TTAAAGAACAAGAAGAAGAAGTCGATGATGAAGTTGTCAACGATATGTTTGGGGGCAAAGATCACGTT
TC
TTTGATCTTCATGGGTCATGTCGATGCTGGTAAGTCCACGATGGGTGGTAATCTGCTTTACTTGACCG
GG
TCCGTGGATAAGAGAACTATTGAAAAATATGAAAGAGAAGCTAAGGATGCCGGTAGACAAGGTTGGTA
TT
TGTCATGGGTCATGGATACCAACAAAGAAGAAAGAAATGATGGTAAGACCATTGAAGTTGGTAAAGCT
TA
TTTCGAAACTGAAAAAAGACGTTATACCATTTTGGATGCTCCAGGTCATAAAATGTACGTTTCAGAAA
TG
ATTGGTGGTGCCTCTCAAGCTGATGTTGGTGTCTTAGTAATTTCTGCCAGAAAGGGTGAATATGAAAC
TG
GTTTCGAAAGAGGTGGTCAAACACGTGAACACGCTTTATTGGCCAAAACTCAAGGTGTTAATAAAATG
GT
TGTTGTCGTAAATAAGATGGACGACCCAACTGTCAACTGGTCCAAGGAACGTTACGACCAATGTGTAA
GT
AATGTCAGCAATTTCTTAAAGGCCATTGGTTACAATATCAAGACTGATGTTGTATTCATGCCAGTATC
AG
GTTACAGTGGTGCAAACTTGAAAGAACATGTAAATCCAAAAGAATGTTCATGGTACACCGGGCCAACT
CT
GTTGGAATACTTGGATAAAATGAATCACGTCGACCGTCACATCAATGCTCCATTCATGTTACCTATTG
CT
GCTAAGATGAAAGATCTAGGTACCATTGTAGAAGGTAAGATCGAATCCGGTCACATAAAGAAGGGTCA
AT
CAACTCTTTTAATGCCTAACAAAACATCAGTGGAAATTCAAAATATTTACAATGAAACTGAAAACGAA
GT
TGATATGGCTATGTGTGGTGAACAAGTTAAATTAAGAATTAAAGGTGTTGAAGAAGAAGATATTTCAC
CA
GGGTTCGTCTTGACATCACCAAAGAACCCAATTAAGAGTGTTACCAAGTTTGTAGCTCAAATTGCCAT
TG
TTGAACTGAAATCTATTATAGCTGCTGGGTTTTCATGTGTTATGCACGTTCACACAGCCATTGAAGAA
GT
TCATATCGTTAAGCTATTGCACAAGTTAGAAAAGGGTACTAACCGTAAATCGAAGAAGCCACCTGCTT
TT
GCTAAGAAGGGTATGAAGGTCATTGCTGTTCTAGAGACTGAAGCTCCAGTTTGTGTCGAGACTTACCA
AG
ACTACCCTCAATTGGGTAGATTCACTTTAAGAGACCAAGGTACCACAATCGCCATTGGTAAGATTGTT
AA
GATTGCTGAATAA
>SUP35_Seub_CBS12357_chr_II_IV_DF968535
ATGTCTGATCCAAACCAAGGTAACAATCAGCAAAACTATCAACAGTACGGTCAAAACTTCAACCAACA
GC
AAGGTAACAACAAATTTCAAGGTTACCAAGCTTACAATGCTCAAGCCCAACAACCTGCAGGTGGCTAT
TA
CCAAAACCCCCAAGGTTACGCTGGCTACCAACAGGGCGGTTATGACCAACAATTTAACCCGGAAGTAG
GT
TACCAACAACAATACAACGCCCAAGGTGGTTACCAACAACAGTTCAATCCACAAGGTGGCCGTGGCAA

TT
ACAAGAACTTCAACTACAACAACAGCCAACAGGGATTCCAAGCTGGCTTTCAACCACAATCTCAAGGA
AT
GTCTTTGAACGACTTCCAAAAGCAACAAAAACAAACTGCTCCTAAGCCAAAGAAGACTTTAAAACTTG
TT
TCAAGTTCCGGTATCAAGTTAGCTAATGCGACCAAGAAGGTCGACACAAAGCCTGTCGAAACTGAGAA
GA
AAGAGGAAGAAAAGCCTACCGAAACCAAAGAACCAGCCAAGGTCGAAGAATCAATCAAAGATGTGGAA
AC
TCCAGCCAGCGCTGAAGAAAAGAAGGAGGTGAATTCTGAATTACCAAAGGTCGAAGATTTGAAAATAT
CC
GAATCGAACGATAACAGCAAGCCTGCTAACCCTATCAACGCCGATGCCTTGATCAAAGAACAAGAAGA
TG
AAGTAGACGATGAAGTTGTTAATGATATGTTCGGAGGTAAAGATCACGTTTCTTTAATTTTCATGGGT
CA
CGTCGATGCTGGTAAGTCCACTATGGGTGGTAATCTACTATATTTGACTGGCTCTGTGGACAAGAGAA
CC
ATTGAAAAATATGAAAGAGAAGCTAAAGATGCTGGTAGACAAGGTTGGTATTTGTCTTGGGTCATGGA
CA
CCAATAAAGAAGAAAGAAACGACGGTAAGACCATTGAGGTCGGTAAGGCTTATTTCGAAACCGAAAAA
AG
ACGTTACACCATTCTGGATGCACCAGGTCATAAAATGTACGTTTCCGAAATGATTGGTGGTGCCTCTC
AA
GCCGATGTTGGTGTTCTAGTCATTTCTGCCAGAAAGGGTGAATACGAAACCGGTTTCGAAAGAGGTGG
TC
AAACACGTGAACACGCCTTATTGGCCAAGACCCAAGGTGTTAATAAGATGATCGTTGTCGTAAATAAG
AT
GGATGATCCTACTGTCAAATGGTCCAAAGAACGTTACGACCAATGTGTAGGTAACGTCAGCAATTTCT
TA
AAGGCCATCGGTTACAATATAAAGACTGATGTTATATTCATGCCAGTATCAGGTTACAGTGGTGCTAA
CT
TGAAAGAACATGTAGATCCAAAGGAATGTTCATGGTACACCGGCCCAACTTTACTGGAATACTTGGAC
AA
AATGACCCACGTTGACCGTCGTATCAACGCTCCATTCATGTTACCTATTGCTGCTAAAATGAAGGATT
TA
GGTACAATCGTAGAAGGTAAGATTGAATCTGGTCACATCAAGAAGGGCCAATCCACTTTATTGATGCC
TA
ATAAGACCACTGTGGAAATCCAAAACATCTATAACGAAACTGAAACCGAAGTCGATATGGCCATGTGT
GG
TGAACAAGTCAAACTAAGAATTAAAGGTGTCGAAGAAGACGACATTTCACCAGGGTTTGTGTTGACAT
CC
CCAAAGAACCCAATTAAGAATGTTACCAAGTTTGTGGCGCAAATTGCCATTGTCGAATTAAAGTCCAT
TA
TAGCCGCCGGATTTTCATGTGTTATGCATGTTCACACAGCCATTGAAGAGGTTCATATTGTTAAACTA
CT
GCATAAGTTAGAAAAGGGTACAAATCGTAAATCGAAGAAGCCACCTGCTTTTGCTAAGAAGGGTATGA
AG
GTCATTGCTGTTTTAGAGACTGAAGCTCCAGTTTGCGTCGAGACTTACCAAGATTATCCACAGTTGGG
TA
GATTTACACTTAGAGACCAAGGTACCACAATTGCCATTGGTAAGATTGTCAAGATTGCTGAATGA

In [5]:

```
#What sequences were used for analysis? Provide type of biopolymer, organism, ge
ne name.

#def find_organism(file):

    # get seqs from fasta file
#    for record in SeqIO.parse(file,"fasta"):
        # run BLAST
#        blastResult = NCBIWWW.qblast("blastn", "nt", record.seq)
        # get first hit
#        blastRecord = NCBIXML.read(blastResult)
#        firstHit = blastRecord.alignments[0]
        # get hit's gi number
#        title = firstHit.title
#        gi = title.split("|")[1]
        # search NCBI for the gi number
#        ncbiResult = Entrez.efetch(db="nucleotide", id=gi, rettype="gb", retmod
e="text")
 #       ncbiResultSeqRec = SeqIO.read(ncbiResult,"gb")
#        first_organism = blastRecord.descriptions[0]

#find_organism("/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SU
P35_10seqs.fa")
```

1) What sequences were used for analysis? Provide type of biopolymer, organism, gene name.

In [7]:

```
fasta_record = SeqIO.parse("/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_10seqs.fa", format='fasta')
pattern = 'ALIGNMENTS\n'
for count, record in enumerate(fasta_record):
    blastResult = list(NCBIWWW.qblast("blastn", "nt", record.seq, format_type = 'text'))
    start = blastResult.index(pattern)
    origin = blastResult[start+1]
    print(f'Sequence number {count} is : {origin}')
```

```
Sequence number 0 is : >CP021242.1 Kluyveromyces lactis strain GG799
chromosome D, complete sequence

Sequence number 1 is : >NM_211584.2 Eremothecium gossypii ATCC 10895
AGL145Wp (AGOS_AGL145W), partial

Sequence number 2 is : >CP036483.1 Saccharomyces cerevisiae strain y
SR128 chromosome IV, complete

Sequence number 3 is : >CP020279.1 Saccharomyces paradoxus strain YP
S138 chromosome IV, complete

Sequence number 4 is : >LT986465.1 Saccharomyces jurei genome assemb
ly, chromosome: IV

Sequence number 5 is : >LR215954.1 Saccharomyces kudriavzevii strain
CR85 genome assembly, chromosome:

Sequence number 6 is : >CP020160.1 Saccharomyces cerevisiae strain D
BVPG6765 chromosome IV, complete

Sequence number 7 is : >CP004661.2 Saccharomyces cerevisiae YJM193 c
hromosome IV sequence

Sequence number 8 is : >CP020279.1 Saccharomyces paradoxus strain YP
S138 chromosome IV, complete

Sequence number 9 is : >CP030946.1 Saccharomyces eubayanus strain CB
S12357 chromosome II, complete
```

2) Running different alignment algorithms (clustalw, muscle, clustalO, mafft and prank) for 10 DNA sequences (SUP35_10seqs.fa)

In [6]:

```
#Create comparison table with running time

from time import process_time
```

In [15]:

```
#sudo apt-get install clustalw


import Bio.Align.Applications
from Bio.Align.Applications import ClustalwCommandline

clustalw = r"/usr/bin/clustalw"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_10seqs.fa"
out_file = r"./clustalw/Clustal10.fasta"
start = process_time()
cline_clustalw = ClustalwCommandline("clustalw", infile = in_file, outfile = out_file  )
stop = process_time()
t1 = stop - start
print(t1, "seconds")
```

```
0.002311546999999914 seconds
```

In [16]:

```
# Muscle alignment

from Bio.Align.Applications import MuscleCommandline

muscle = r"/usr/bin/muscle"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_10seqs.fa"
out_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/process_alignments/muscle/Muscle10.fasta"
start = process_time()
cline_Muscle = MuscleCommandline(muscle, input = in_file, out = out_file)
stop = process_time()
t2 = stop - start
print(t2, "seconds")
```

```
0.0010732939999997804 seconds
```

In [20]:

```
# Prank alignment
#sudo apt-get update -y
#sudo apt-get install -y prank

from Bio.Align.Applications import PrankCommandline
import os

in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_
10seqs.fa"
start = process_time()
prank_cline = PrankCommandline(d = in_file,
                               o ="aligned",
                               f = 8, # FASTA output
                               notree = True, noxml = True)
prank_cline()
stop = process_time()
t3 = stop - start
print(t3, "seconds")
```

```
0.011364158000000124 seconds
```

In [22]:

```
# Mafft alignment:

#sudo apt install mafft

from Bio.Align.Applications import MafftCommandline

mafft = r"/usr/bin/mafft"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_
10seqs.fa"
start = process_time()
mafft_cline = MafftCommandline(mafft, input=in_file)
stdout, stderr = mafft_cline()
with open("mafft10.fasta", "w") as handle:
    handle.write(stdout)
mafft_cline()
stop = process_time()
t4 = stop - start
print(t4, "seconds")
```

```
0.014952075999999925 seconds
```

In [25]:

```
# ClustalOmega alignment:
#sudo apt-get install clustalo

from Bio.Align.Applications import ClustalOmegaCommandline

clustalo = r"/usr/bin/clustalo"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/homework_3/SUP35_
10seqs.fa"
out_file = r"./clustalo/ClustalOmega.fasta"
start = process_time()
cline_ClustalO = ClustalOmegaCommandline(clustalo, infile=in_file, outfile=out_f
ile, verbose=True, auto=True)
stop = process_time()
t5 = stop - start
print(t5, "seconds")
```

```
0.0008695830000000626 seconds
```

running time comparison table

In [29]:

```
from prettytable import PrettyTable
x = PrettyTable()
x.field_names = ['algorithm', 'Running_time']
x.add_row(['ClustalW', t1])
x.add_row(['Muscle', t2])
x.add_row(['Prank', t3])
x.add_row(['Mafft', t4])
x.add_row(['ClustalO', t5])
print(x)
```

```
+-----------+-----------------------+
| algorithm |      Running_time     |
+-----------+-----------------------+
|  ClustalW |  0.002311546999999914 |
|   Muscle  | 0.0010732939999997804 |
|   Prank   |  0.011364158000000124 |
|   Mafft   |  0.014952075999999925 |
|  ClustalO | 0.0008695830000000626 |
+-----------+-----------------------+
```

Based on running time I can tell that clustalo if faster that other algorithms!

the question that arises here is I have read muscle is more accurate than T-Coffee and faster than Clustal-W! but why results does not show this!

comments on the DNA alignment quality for the 5 algorithms. Which algorithm is better to use?

Clustalo shows better results I think because it contain less gaps compared with other methods.

But I read this text about alignments:

Muscle has a better theoretical basis than Clustal. Clustal is highly sensitive to the order in which you list the sequences, because its search algorithm always depends on the order, especially the first sequence. Muscle takes an iterative approach, which is a plus. The advantage of an iterative approach is that it can search through a variety of alignments and phylogenies, make changes to them, test if it's an improvement or not, and continue onward until it has a more optimal solution.

3) What is wrong with the alignment of SUP35_10seqs_strange_aln.fa and how to fix it?

In [56]:

```
#Converting a file of sequences to their reverse complements


from Bio import SeqIO
sequences = (seq.reverse_complement(id="rc_"+seq.id, description = "reverse comp
lement") \
           for seq in SeqIO.parse("/home/sedreh/ITMO/semester3/Molecular_phyloge
netic/homework_3/SUP35_10seqs_strange_aln.fa", "fasta"))
SeqIO.write(sequences, "rev_comp.fasta", "fasta")
```

Out[56]:

```
10
```

4) Obtain amino acid sequences for these data. Repeat p.2 on amino acid sequences.

To get amino acids we need to Tranlate DNA to protein:

https://github.com/prestevez/dna2proteins (https://github.com/prestevez/dna2proteins)

I used python code and got protein sequence from this command in terminal:

sample command: python dna2proteins.py -i sequences.fa -o proteins.fa -p

python /home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/ -i /home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/SUP35_10seqs.fa -o /home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/

In [ ]:

```
#second approach

file = open('/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/SUP35_10seqs.fa', 'r')
dna = file.read()

#print ("DNA Sequence: ", dna)

# DNA codon table
protein = {"TTT" : "F", "CTT" : "L", "ATT" : "I", "GTT" : "V",
           "TTC" : "F", "CTC" : "L", "ATC" : "I", "GTC" : "V",
           "TTA" : "L", "CTA" : "L", "ATA" : "I", "GTA" : "V",
           "TTG" : "L", "CTG" : "L", "ATG" : "M", "GTG" : "V",
           "TCT" : "S", "CCT" : "P", "ACT" : "T", "GCT" : "A",
           "TCC" : "S", "CCC" : "P", "ACC" : "T", "GCC" : "A",
           "TCA" : "S", "CCA" : "P", "ACA" : "T", "GCA" : "A",
           "TCG" : "S", "CCG" : "P", "ACG" : "T", "GCG" : "A",
           "TAT" : "Y", "CAT" : "H", "AAT" : "N", "GAT" : "D",
           "TAC" : "Y", "CAC" : "H", "AAC" : "N", "GAC" : "D",
           "TAA" : "STOP", "CAA" : "Q", "AAA" : "K", "GAA" : "E",
           "TAG" : "STOP", "CAG" : "Q", "AAG" : "K", "GAG" : "E",
           "TGT" : "C", "CGT" : "R", "AGT" : "S", "GGT" : "G",
           "TGC" : "C", "CGC" : "R", "AGC" : "S", "GGC" : "G",
           "TGA" : "STOP", "CGA" : "R", "AGA" : "R", "GGA" : "G",
           "TGG" : "W", "CGG" : "R", "AGG" : "R", "GGG" : "G"
           }

protein_sequence = ""

# Generate protein sequence
for i in range(0, len(dna)-(3+len(dna)%3), 3):
    if protein[dna[i:i+3]] == "STOP" :
        break
    protein_sequence += protein[dna[i:i+3]]

# Print the protein sequence
print ("Protein_Sequence: ", protein_sequence)
```

In [10]:

```
# Muscle:

from Bio.Align.Applications import MuscleCommandline
muscle_exe = r"/usr/bin/muscle"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/protein/protein.fasta"
out_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/protein/Muscle10p.fasta"

start = process_time()
cline = MuscleCommandline(muscle_exe, input=in_file, out =out_file)
cl = str(cline)
stdout, stderr = cline()
stop = process_time()
t_p1 = stop - start
print(t_p1, "seconds")
```

```
0.01476589099999992 seconds
```

In [11]:

```
# ClustalW:
import Bio.Align.Applications
from Bio.Align.Applications import ClustalwCommandline

clustalw = r"/usr/bin/clustalw"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/protein/protein.fasta"
out_file = r"./clustalw/Clustal10p.fasta"
start = process_time()
cline_clustalw = ClustalwCommandline("clustalw", infile = in_file, outfile = out_file  )
stop = process_time()
t_p2 = stop - start
print(t_p2, "seconds")
```

```
0.0029270630000000075 seconds
```

In [20]:

```
# Prank:


from Bio.Align.Applications import PrankCommandline
import os

in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_par
t1/homework_3/Biopython_process/protein/protein.fasta"
start = process_time()
prank_cline = PrankCommandline(d = in_file,
                               o ="aligned",
                               f = 8, # FASTA output
                               notree = True, noxml = True)
prank_cline()
stop = process_time()
t_p3 = stop - start
print(t_p3, "seconds")
```

```
0.007622177000000008 seconds
```

In [14]:

```
# mafft:


from Bio.Align.Applications import MafftCommandline

mafft_exe = r"/usr/bin/mafft"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_par
t1/homework_3/Biopython_process/protein/protein.fasta"

start = process_time()
mafft_cline = MafftCommandline(mafft_exe, input=in_file)
stdout, stderr = mafft_cline()
with open("mafft10p.fasta", "w") as handle:
    handle.write(stdout)
mafft_cline()
stop = process_time()
t_p4 = stop - start
print(t_p4, "seconds")
```

```
0.012345296000000117 seconds
```

In [17]:

```
# ClustalOmega:

from Bio.Align.Applications import ClustalOmegaCommandline
clustalo_exe = r"/usr/bin/clustalo"
in_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/protein/protein.fasta"
out_file = r"/home/sedreh/ITMO/semester3/Molecular_phylogenetic/phylogenetics_part1/homework_3/Biopython_process/protein/ClustalO10p.fasta"

start = process_time()
cline = ClustalOmegaCommandline(clustalo_exe, infile=in_file, outfile=out_file,
force = True)
stdout, stderr = cline()
cline()
stop = process_time()
t_p5 = stop - start
print(t_p5, "seconds")
```

```
0.0071628729999999585 seconds
```

In [21]:

```
# Comparison table of algorithms running time for protein:

from prettytable import PrettyTable
x = PrettyTable()
x.field_names = ['algorithm', 'Running_time']
x.add_row(['ClustalW', t_p2])
x.add_row(['Muscle', t_p1])
x.add_row(['Prank', t_p3])
x.add_row(['Mafft', t_p4])
x.add_row(['ClustalO', t_p5])
print(x)
```

```
+-----------+-----------------------+
| algorithm |      Running_time     |
+-----------+-----------------------+
|  ClustalW | 0.0029270630000000075 |
|   Muscle  |  0.01476589099999992  |
|   Prank   |  0.007622177000000008 |
|   Mafft   |  0.012345296000000117 |
|  ClustalO | 0.0071628729999999585 |
+-----------+-----------------------+
```

5) Optional Repeat p. 2 on the alignment of 250 DNA sequences (SUP35_250seqs.fa). Has our choice of algorithm changed?

I have done it and attached files.

6) How to add to the alignment another sequence (SUP35_1addseq.fsa) using mafft and muscle?

adding new sequence to the alignment using mucsle

http://www.drive5.com/muscle/muscle_userguide3.8.htr (http://www.drive5.com/muscle/muscle_userguide3.8.ht

To add a sequence to an existing alignment, use profile-profile alignment with the new sequence as a profile.

For example, if you have an existing alignment Muscle10.fasta and want to add a new sequence in SUP35_1addseq.fsa, use the following commands: muscle -profile -in1 existing_aln.afa -in2 new_seq.fa -out combined.afa

adding new sequence to the alignment using mafft

https://mafft.cbrc.jp/alignment/software/addsequences.l (https://mafft.cbrc.jp/alignment/software/addsequences

% mafft --add new_sequences --reorder existing_alignment > output

% mafft --addfragments fragments --reorder --thread -1 existing_alignment > output

7) How to add two more sequences (SUP35_2addseqs.fsa), pre-aligning them with the mafft or muscle?

In [ ]:

8) #Try to run Gblocks (for alignment of 10 and 250 sequences). What percent of alignment remains after starting Gblocks with strict and non-strict parameters(specify parameters)?

http://molevol.cmima.csic.es/castresana/Gblocks_server.html (http://molevol.cmima.csic.es/castresana/Gblocks_server.html)

```
######## strict parameters #######

    ######  86% of the original 2153 positions#####
```

Parameters used Minimum Number Of Sequences For A Conserved Position: 6 Minimum Number Of Sequences For A Flanking Position: 8 Maximum Number Of Contiguous Nonconserved Positions: 4 Minimum Length Of A Block: 10 Allowed Gap Positions: None

Flank positions of the 15 selected block(s) Flanks: [33 44] [67 84] [118 172] [205 224] [232 251] [311 331] [338 371] [399 421] [425 470] [474 487] [492 610] [654 697] [708 720] [722 740] [743 2150]

New number of positions in ClustalOmega.fasta-gb: 1866 (86% of the original 2153 positions)

```
######## non-strict parameters ########

    ###### 94% of the original 2153 positions #######
```

Parameters used Minimum Number Of Sequences For A Conserved Position: 6 Minimum Number Of Sequences For A Flanking Position: 6 Maximum Number Of Contiguous Nonconserved Positions: 8 Minimum Length Of A Block: 5 Allowed Gap Positions: With Half

Flank positions of the 16 selected block(s) Flanks: [1 8] [33 62] [67 90] [118 178] [182 190] [194 224] [231 331] [338 371] [377 470] [474 487] [492 610] [614 631] [644 697] [701 720] [722 740] [743 2150]

New number of positions in ClustalOmega.fasta-gb: 2044 (94% of the original 2153 positions)