

ΑΠΑΛΛΑΚΤΙΚΗ ΕΡΓΑΣΙΑ ΒΙΟΠΛΗΡΟΦΟΡΙΚΗΣ 6^{ΟΥ} ΕΞΑΜΗΝΟΥ 2021-2022

Ομάδα εκπόνησης εργασίας:

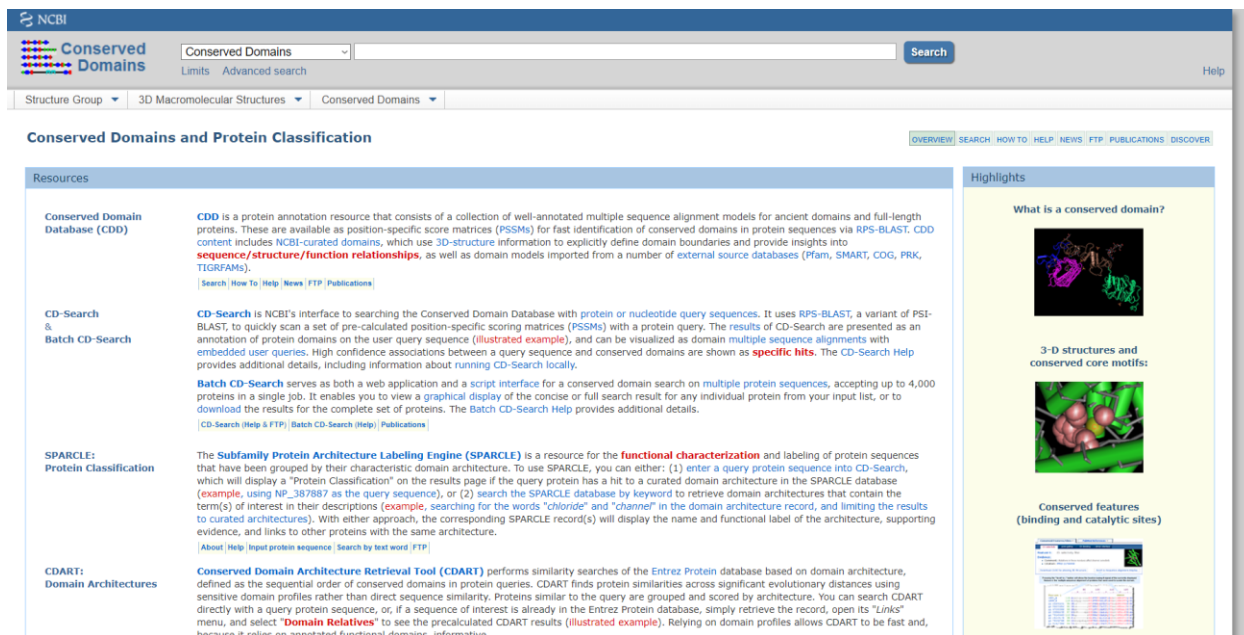
Αγγελική Αποστόλου Π19015, Ευρυδίκη Βαλή-Σαράφογλου Π19023, Αντώνιος Καλλίας-Βερβεγιώτης Π19056

Θέμα 1^ο:

Η άσκηση προς επίλυση είναι η άσκηση 7.2 από το βιβλίο "Βιοπληροφορική και Λειτουργική Γονιδιωματική". Ζητούμενο είναι η πραγματοποίηση φυλογενετικών αναλύσεων με τη χρήση του λογισμικού MEGA. Τα βήματα για την εκπόνηση της συγκεκριμένης άσκησης όπως περιγράφονται από την εκφώνηση είναι:

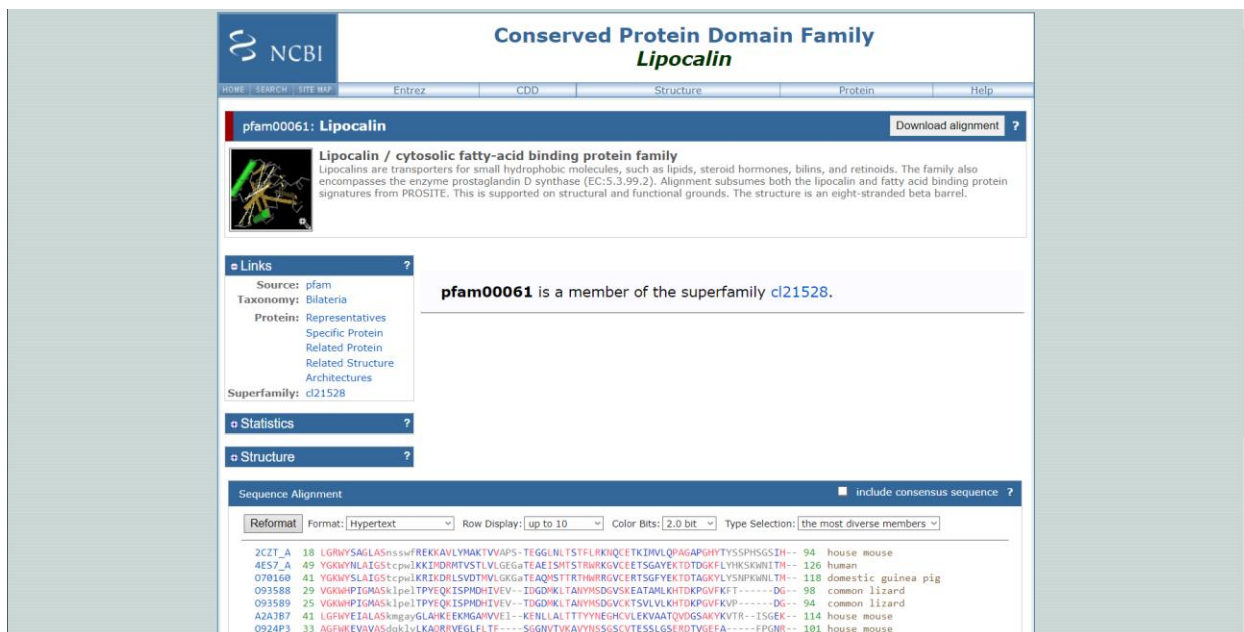
1. Μετάβαση στη βάση δεδομένων των συντηρημένων δομικών επικρατειών (<https://www.ncbi.nlm.nih.gov/Structure/cdd/cddsrv.cgi?uid=395015>) στο NCBI
2. Εισαγωγή του όρου λιποκαλίνες (lipocalins) ή άλλου ονόματος οικογένειας
3. Επιλογή μορφής αρχείου mFasta και στη συνέχεια επιλογή «Reformat». Το αποτέλεσμα είναι μια πολλαπλή στοίχιση αλληλουχιών. Αντιγραφή του αποτελέσματος σε έναν επεξεργαστή κειμένου και απλοποίηση των ονομάτων των αλληλουχιών
4. Εισαγωγή του αρχείου στο MEGA. Στοίχιση των αλληλουχιών και αποθήκευση σε μορφές αρχείων .mas και .meg
5. Επιλογή Phylogeny>Construct>Test>(Φυλογένεση>Κατασκευή/Δοκιμή) για την δημιουργία δέντρων με τις μεθόδους ένωσης γειτόνων, μέγιστης πιθανοφάνειας ή άλλες
6. Για κάθε δέντρο που δημιουργείται, ανάγνωση της σχετικής λεζάντας. Δοκιμή των εργαλείων δέντρων
7. Πραγματοποίηση bootstrapping. Προσδιορισμός των συστάδων κλάδων που έχουν χαμηλά επίπεδα στήριξης. Γιατί συμβαίνει αυτό;

Βήμα 1: Μετάβαση στη βάση δεδομένων των συντηρημένων δομικών επικρατειών (<https://www.ncbi.nlm.nih.gov/Structure/cdd/cddsrv.cgi?uid=395015>) στο NCBI



The screenshot displays the NCBI Conserved Domains and Protein Classification page. The top navigation bar includes links for Overview, Search, How To, Help, News, FTP, Publications, and Discover. The main content area is divided into two columns. The left column, titled 'Resources', lists several tools: Conserved Domain Database (CDD), CD-Search & Batch CD-Search, SPARCLE: Protein Classification, and CDART: Domain Architectures. The right column, titled 'Highlights', features a section 'What is a conserved domain?' with a 3D structure of a protein and a list of '3-D structures and conserved core motifs'. Below this, there is a section 'Conserved features (binding and catalytic sites)' with a table of features.

Βήμα 2: Εισαγωγή του όρου λιποκαλίνες (lipocalins) ή άλλου ονόματος οικογένειας



The screenshot displays the NCBI Conserved Protein Domain Family page for Lipocalin. The page shows the family name 'Lipocalin' and a description of the family. The 'Links' section on the left provides navigation options for Source, Taxonomy, Proteins, and Superfamily. The 'Statistics' and 'Structure' sections are also visible. The 'Sequence Alignment' section at the bottom shows a table of sequence alignments for various species.

Βήμα 3: Επιλογή μορφής αρχείου mFasta και στη συνέχεια επιλογή «Reformat». Το αποτέλεσμα είναι μια πολλαπλή στοίχιση αλληλουχιών. Αντιγραφή του αποτελέσματος σε έναν επεξεργαστή κειμένου και απλοποίηση των ονομάτων των αλληλουχιών

Reformat σε mFasta

Sequence Alignment

Reformat

Format: mFasta

Row Display: up to 10

Color Bits: 2.0 bit

Type Selection: the most diverse members

>gi|116666961|pdb|2CZT|A

-----gsqghdtvqpnfqgdkfLGRWYSAGLASnswfREKKAVLYMAKTVVAP

S-TEGGILNLTSTFLRKINQCEKIMVLQPAGAGHYTYSSPHSGSIH---SVSVVEANYDEYALFSGRTGpGgQDFRMA

LYSRQTQLKDELke-kFTTFSKAQGLTEEDIVFLPQpDkcIqe-----

>gi|482677111|pdb|4ES7|A

mgsshhhhhhssglvprgshmasmtggqgmrgsdniqvqenfnisriYGWYNLAIGStcpwlKKIMDRMTVSTLVLE

GaTEAEISMTSTRWRKGVCEETSGAYEKTDGKFLYHCKWILTH---ESYVVTNYDEYALFLLKFSRHgPTITAK

LYGRAPQLRETLlq-dFRVVAQGVGIPEDSIFTMADRGecvpgqe-----

>gi|81886784|sp|070160|070160_CAVPO

-----mgaeall111gsclaisvnpvltlppdiqgqenfdesrmYGWYSLAIGStcpwlKRIKRLSVDTNVLGK

GA TEAQMSTRTTHWRKGVCEETSGAYEKTDGKFLYHCKWILTH---ESYVVTNYDEYATVLTKKFSRHgPTITAK

LYGREPQLRDSLlq-gFREMAISVGIPEDSIFTMANRGeicpgeipqptpalrarravllqedeqsgagpvtldftkke

dsqqlghsagpcclglfkryfyngssmaceifhyggclngnfnfsekeclqtrtvaacnlpivpgcagsaqlwafda

kgkcvrfttyggcgqngnkyfsekeckeycgagdgdeellrfsn

>gi|82123701|sp|093588|093588_LACVV

-----lvfgmtptyifpvsadipvpvfnfdpqtVGKwHPITGMASKlpeITPYEQKISPMOHIVEV

--IDGDKLITANYSDGVSKEATAMLKHTDKPGVFKFT---DG---EVHVLVDVDFEKYIMLYVKKS---SHEALF

LSARGPVEDDIke-kFKKLVLQSFPEANIKYFNAEQctptaa-----

>gi|82071154|sp|093589|093589_LACVV

-----mtpdyifpvsadipvpvfnfdpqtVGKwHPITGMASKlpeITPYEQKISPMOHIVEV

--IDGDKLITANYSDGVSKEATAMLKHTDKPGVFKFT---DG---EVHVLVDVDFEKYIMLYVKKS---SHEALF

LSARGSTGGDDIke-kFKKLVLQSFPEANIKYFNAEQctptaa-----

>gi|190359792|sp|A2A3B7.1|LCN5_MOUSE

-----mcsvvarhmesimlftl1glcvglagteaavvkfdvknfLGFWYETALASKmgayGLAHKEEKMGAMVVEI

--KENLLALTTYYNIEGHCVLEKVAATQVDGSAKYKVT---ISGK---EVVVVATDYMTYTVIDITSLVA-GaVHRAMK

LYSRSLDWNNGEAln-nFQKIALKHGFSETDIHLKHDLTcvnalqsgql-----

>gi|62286893|sp|Q924P3.1|LCN8_MOUSE

-----lmaaeITenunaffTlvtflTasgpfenatunakTAGDSEFVNAKAdnlvliKANDRUCGIETL---

Λήψη στοίχισης

NCBI

Conserved Protein Domain Family

Lipocalin

HOME

SEARCH

SITE MAP

Entrez

CDD

Structure

Protein

Help

pfam00061: Lipocalin

Download alignment



Lipocalin / cytosolic fatty-acid binding protein family

Lipocalins are transporters for small hydrophobic molecules, such as lipids, steroid hormones, bilins, and retinoids. The family also encompasses the enzyme prostaglandin D synthase (EC:5.3.99.2). Alignment subsumes both the lipocalin and fatty acid binding protein signatures from PROSITE. This is supported on structural and functional grounds. The structure is an eight-stranded beta barrel.

Links

Source: pfam

Taxonomy: Bilateria

Protein: Representatives

Specific Protein

Related Protein

Related Structure

Architectures

Superfamily: cl21528

Statistics

Structure

Sequence Alignment

include consensus sequence

Reformat

Format: mFasta

Row Display: up to 10

Color Bits: 2.0 bit

Type Selection: the most diverse members

>gi|116666961|pdb|2CZT|A

-----gsqghdtvqpnfqgdkfLGRWYSAGLASnswfREKKAVLYMAKTVVAP

S-TEGGILNLTSTFLRKINQCEKIMVLQPAGAGHYTYSSPHSGSIH---SVSVVEANYDEYALFSGRTGpGgQDFRMA

LYSRQTQLKDELke-kFTTFSKAQGLTEEDIVFLPQpDkcIqe-----

Αντιγραφή των αλληλουχιών

>gi|116666961

-----gsqghdtvqpnfqdkfLG-RWYSAGLASnss-wfREKKAVLYMAKTVV
APS-TEGGLNLTSTF---L-----RKN-Q--CETKIMVLQPA--GAPGHYTYSSPH---SGSIH---SVSVV-EANY
DEYALLFSRGT-KGpGq-DFRMTLYSRTQTLKDELke-kFTTFSKAQ--GLTEED-IVFLPQPDkciqe-----

>gi|82214882

-----mqatlslglallgalhaqnsipvqadfqdkLAG-RWYSIGLASnsn-wfKDKKHLLKMCTTDI
AVT-ADGNMEVTSTY---P-----KGE-Q--CEKRNSLYIRT--EQPGRFSYTNPR---WGSNH---DIRVV-ETNY
DEYALVATQIS-KStG--SSNMVLLYSRTKEVAPQRle-rFMQFSQEQ--GLKDEE-ILILPQTDkcmadaa-----

>gi|131649

-----mkyaqyvflasifsaveyslaqtcavdsfsvkdnfdpkryAG-KWYALAKKD-----PEGLFLQDNISAEY
TVE-EDGTMTASSKGrvkL-----FGF-WviCADMAAQYTVdpTTPAKMYMTYQGLasYLSSGg-dNYWVI-DTDY
DNYAITYACRS1KedGscDDGYSLIFSRNPRGLPPA----IQRIVRQKqeEICMSG-QFQPVLQSGac-----

>gi|75067994

-----maasrglwmgllvllgvlglvltqraqdpvsvqpefqdkfLG-RWFTAGLASnss-wfREKKAALSMCRSTV
APT-EEGALNITSTF---L-----RKN-Q--CETRLLLLQPA--GRPGRYTYTSPH---WGSTY---SVWVV-DTDY
KEFALLYSEGA-KGpGq-DFRMTLYSRSQTPGAELkq-kFMAFCKAQ--GFTEDI-VVFLPRNDkcmeeqd-----

>gi|2497698

-----maalhtlwmgllvllgvlglvltqraqaqsrvqpnfqdkfLG-RWFTSGLASnss-wfREKKNALSMCISVV
APS-AEGGLNLTSTF---L-----RKD-Q--CETRLLLLRPA--ETPGCYSYTSPPH---WGSTH---DVWVV-ATDY
EEYALLYTAGT-KSpGq-DFHMTLYSRTQTPRAEVke-kFSTFAKTR--GFTEDA-IVFLPKTERcmeehr-----

>gi|2497700

-----matpsslwlgllgltglvltqtpaqaslpnfqdkfLG-RWFTSGLASnss-wfLEKKKVLSMCKSLV
APA-PDGGFNLTSTF---L-----RKD-Q--CVTRTLMLRPA--GPPGCYSYTSPPH---GGSNL---EVSVV-ETDY
KNYALLHTESG-PSpGp-AFRMTLYSRSQAPGAAVre-kFTAFAKAR--GFTEDG-IVFLPRNEkcleeh-----

>gi|3914330

-----matpnrlwmallllgvlglvltqpapaqaalqpnfeedkLG-RWFTSGLASnss-wfLEKKKVLSMCKSVV
APA-ADGGLNLTSTF---L-----RKD-Q--CETRLLLLRPA--GPPGCYSYTSPPH---WSSTH---EVSVA-ETDY
ETYALLYTEGV-RGpGq-DFRMTLYSRSQNPRAEVke-hFTTFAKSL--GFTEEG-IVFLPKTDkcmeehp-----

Επικόλληση σε κειμενογράφο

lipocalins.txt - Notepad

File Edit Format View Help

>gi|116666961

-----gsqghdtvqpnfqdkfLG-RWYSAGLASnss-wfREKKAVLYMAKTWV
APS-TEGGLNLTSTF---L-----RKN-Q--CETKIMVLQPA--GAPGHYTYSSPH---SGSIH---SVSVV-EANY
DEYALLFSRGT-KGpGq-DFRMTLYSRTQTLKDELke-kFTTFSKAQ--GLTEED-IVFLPQPDkciqe-----

>gi|131649

-----mkyaqyvflasifsaveyslaqtcavdsfsvkdnfdpkryAG-KWYALAKKD-----PEGLFLQDNISAEY
TVE-EDGTMNTASSKGrvKL-----FGF-WwICADMAAQYTVpdpTTPAKMYMTYQGLasYLSSGg-dNYWVI-DTDY
DNYAITYACRSIkEDGscDDGYSLIFSRNPRGLPPA----IQRIVRQKqeEICMSG-QFQPVLSgac-----

>gi|2497698

-----maalhtlwmglvllgvlglvqltraqaqvsvrqpfnfqdkfLG-RWFTSGLASnss-wfREKKNALSMCISVV
APS-AEGGLNLTTF---L-----RKD-Q--CETRTLRLRPA--ETPGCYSYTSPh--WGSTH---DVWVV-ATDY
EEYALLYTAGT-KSpGq-DFHMTLYSRTQTPRAEVke-kFSTFAKTR--GFTEDA-IVFLPKTERcmeehr-----

>gi|672885873

-----mathhtlwmglallgvlgdlqaapeaqvsvqpfnfqdkfLG-RWFSAGLASnss-wlREKAAALSMASVV
APA-TDGLNLTSTF---L-----RKN-Q--CETRTMLLQPA--GSLGSYSYRSPH---WGSTY---SVSVV-ETDY
DQYALLYSQGS-KGpGe-DFRMTLYSRTQTPRAELke-kFTAFCKAQ--GFTEDT-IVFLPQTDkcmteq-----

>gi|485601482

-----gsqdstqnlipapslltvplqpdrsdqfRG-RWYVVGLAGna--vqKKTEGSFTMYSTIY
ELQ-ENNSYNVTSIL---V-----RDQ-DqgCRYWIRTFVPS--SRAGQFTLGNMHrypQVQSY---NVQVA-TTDY
NQFAMVFFRKT-SEnKq-YFKI-TLYGRTKELSPELke-rFTRFAKSL--GLKDDN-IIFSVPTDqcidn-----

>gi|528082202

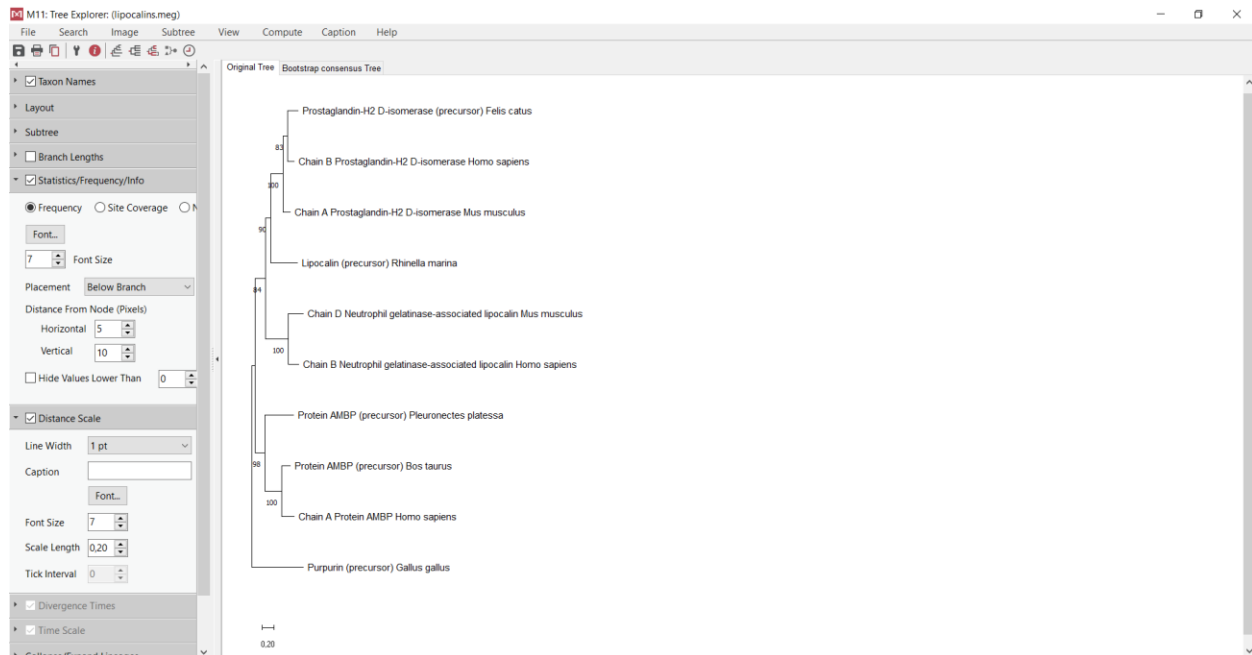
-----gsqdstdlipapplskvplqqnfqdnqfQG-KWYVVGLAGna--ilREDKDPQKMYATIY
ELK-EDKSYNVTSLV---F-----RKK-K--CDYWIRTFVPG--SQPGEFTLGNIKsypGLTSY---LVRVV-STNY
NQHAMVFFKKV-SQnRe-YFKI-TLYGRTKELTSELke-nFIRFSKSL--GLPENH-IVFPVPIDqcidg-----

>gi|266472

Απλοποίηση των ονομάτων των αλληλουχιών

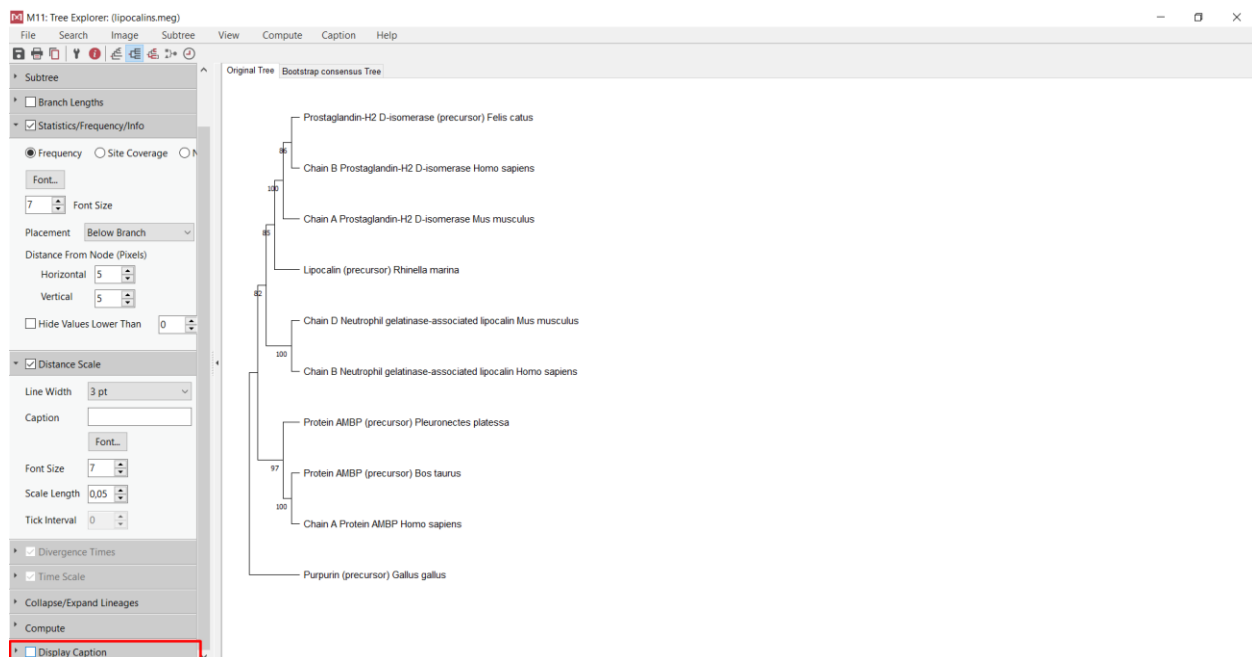
```
>Chain A, Prostaglandin-H2 D-isomerase Mus musculus
-----gsqghdtvqpnfqgdkfLG-RWYSAGLASnss-wFREKKAVLYMAKTVV
APS-TEGGLNLTSTF---L-----RKN-Q--CETKIMVLQPA--GAPGHYTYSSPH---SGSIH---SVSVV-EANY
DEYALLFSRGT-KGpGq-DFRMTLYSRTQTLKDELke-kFTTFSKAQ--GLTEED-IVFLPQPDkciqe-----
-----
>Purpurin (precursor) Gallus gallus
-----mkyaqyvflasifsaveyslaqtcavdsfsvkdnfdpkryAG-KWYALAKKD-----PEGLFLQDNISAEY
TVE-EDGTMATASSKGrvKL-----FGF-wviCADMAAQYVTPdpTTPAKMYMTYQGLasYLSSGg-dNYWVI-DTDY
DNYAITIYACRSIkEdGscDDGYSLIFSRNPRGLPPA----IQRIVRQKqeEICMSG-QFQPVLSgac-----
-----
>Prostaglandin-H2 D-isomerase (precursor) Felis catus
-----maalhtlwmglvllgvlglvltqraqavsrqpnfqgdkfLG-RWFTSGLASnss-wFREKKNALSMCISVV
APS-AEGGLNLTSTF---L-----RKD-Q--CETRTLMLLQPA--ETPGCYSYTSPPH---WGSTH---DVWVV-ATDY
EEYALLYTAGT-KSpGq-DFHMTLYSRTQTPRAEVke-kFSTFAKTR--GFTEDA-IVFLPKTERcmeehr-----
-----
>Chain B, Prostaglandin-H2 D-isomerase Homo sapiens
-----mathhtlwmglallgvlglgdlqaapeaqvsvqpnfqgdkfLG-RWFSAGLASnss-wlREKAAALSMASVV
APA-TDGLNLTSTF---L-----RKN-Q--CETRTMLLQPA--GSLGSYSYRSPH---WGSTY---SVSVV-ETDY
DQYALLYSQGS-KGpGe-DFRMTLYSRTQTPRAELke-kFTAFAKQ--GFTEDT-IVFLPQTDkcmtcq-----
-----
>Chain D, Neutrophil gelatinase-associated lipocalin Mus musculus
-----gsqdstqnlipapslltvpplqpdfrsdqfRG-RWYVVGLAGna--vqKKTEGSFTMYSTIY
ELQ-ENNSYNVTSIL---V-----RDQ-DqgCRYWIRTFVPS--SRAGQFTLGNMHrypQVQSY---NVQVA-TTDY
NQFAMVFFRKT-SEnKq-YFKI-TLYGRTKELSPeLke-rFTRFAKSL--GLKDDN-IIFSVPIDqcidn-----
-----
>Chain B, Neutrophil gelatinase-associated lipocalin Homo sapiens
-----gsqdstsdlipapplskvplqqnfqdnqfQG-KWYVVGLAGna--ilREDKDPQKMYATIY
ELK-EDKSYNVTSLV---F-----RKK-K--CDYWIRTFVPG--SQPGEFTLGNIKsypGLTSY---LVRVV-STNY
NQHAMVFFKKV-SQnRe-YFKI-TLYGRTKELTSELke-nFIRFSKSL--GLPENH-IVFPVPIDqcidg-----
-----
>Lipocalin (precursor) Rhinella marina
```

Βήμα 4 Εισαγωγή του αρχείου στο MEGA.

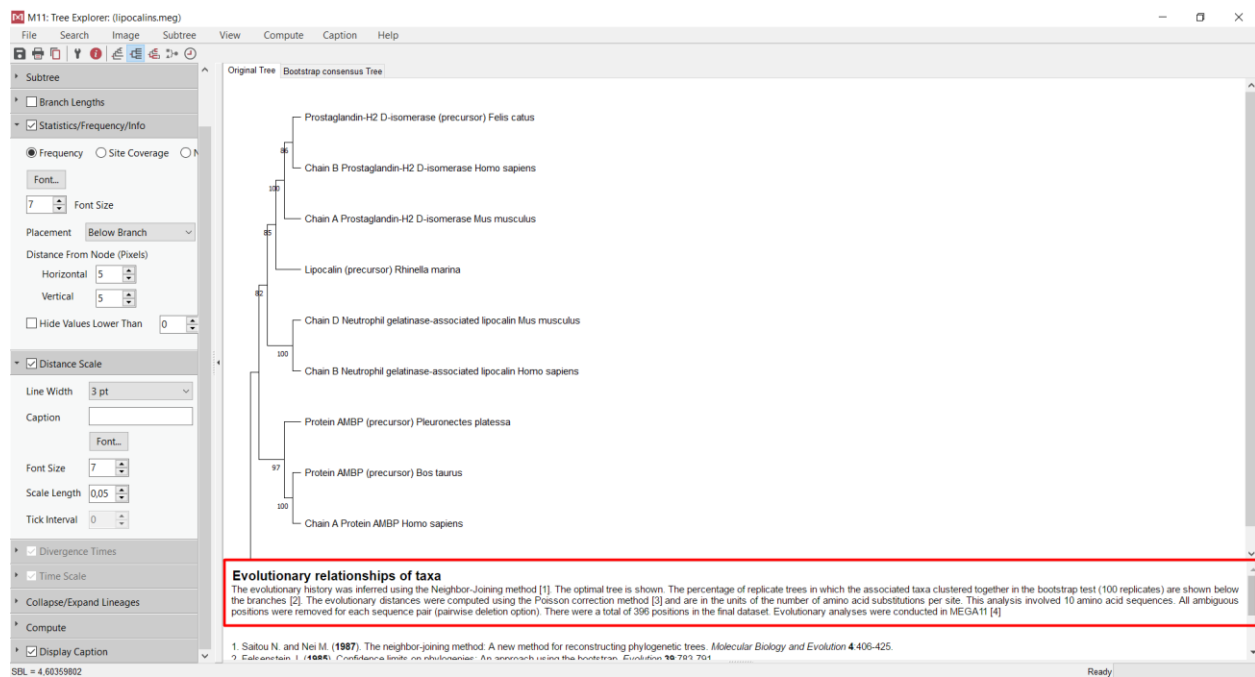


Βήμα 6: Για κάθε δέντρο που δημιουργείται, ανάγνωση της σχετικής λεζάντας. Δοκιμή των εργαλείων δέντρων

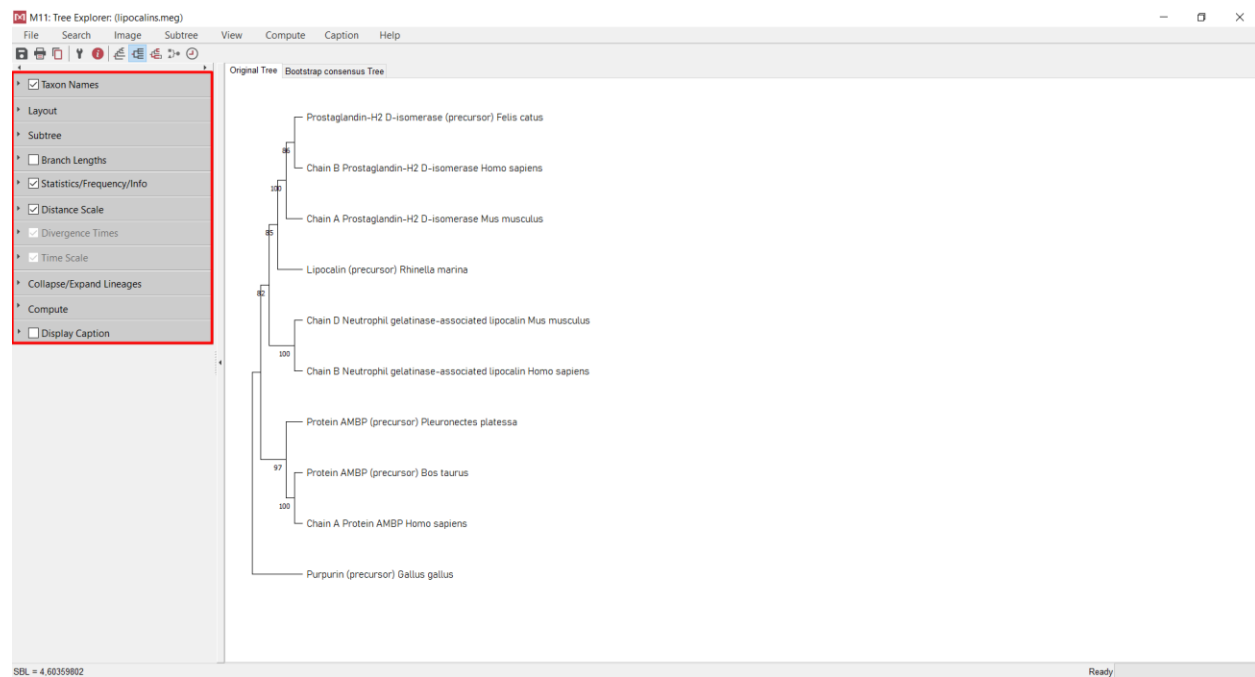
Επιλογή Display Caption για την εμφάνιση της λεζάντας



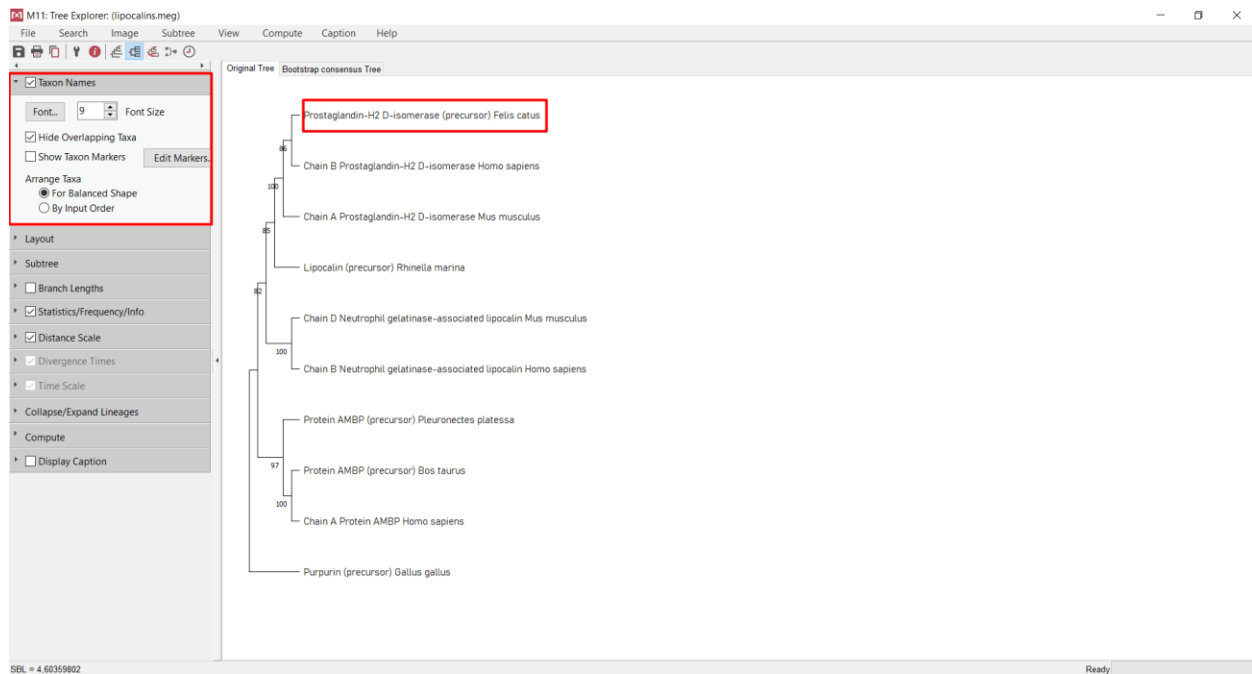
Ανάγνωση λεζάντας



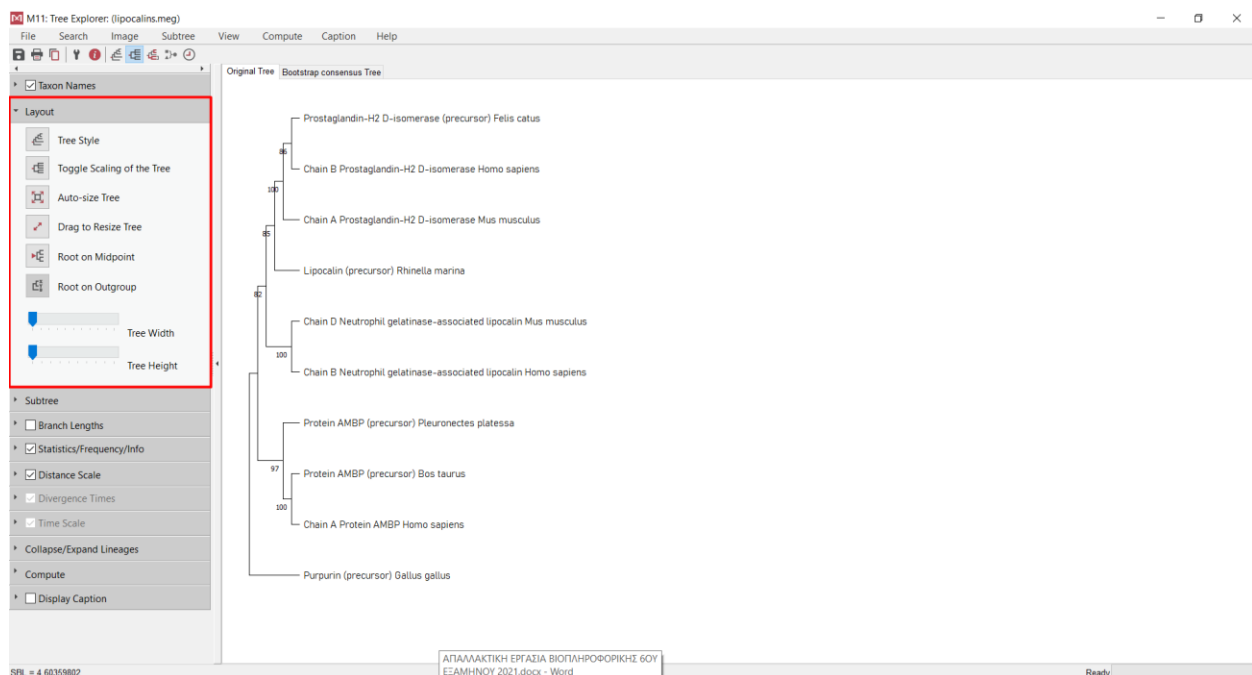
Στο αριστερό μέρος της οθόνης παρατηρούμε την εργαλειοθήκη που περιλαμβάνει τα διάφορα εργαλεία για την επεξεργασία του προβαλλόμενου δέντρου.



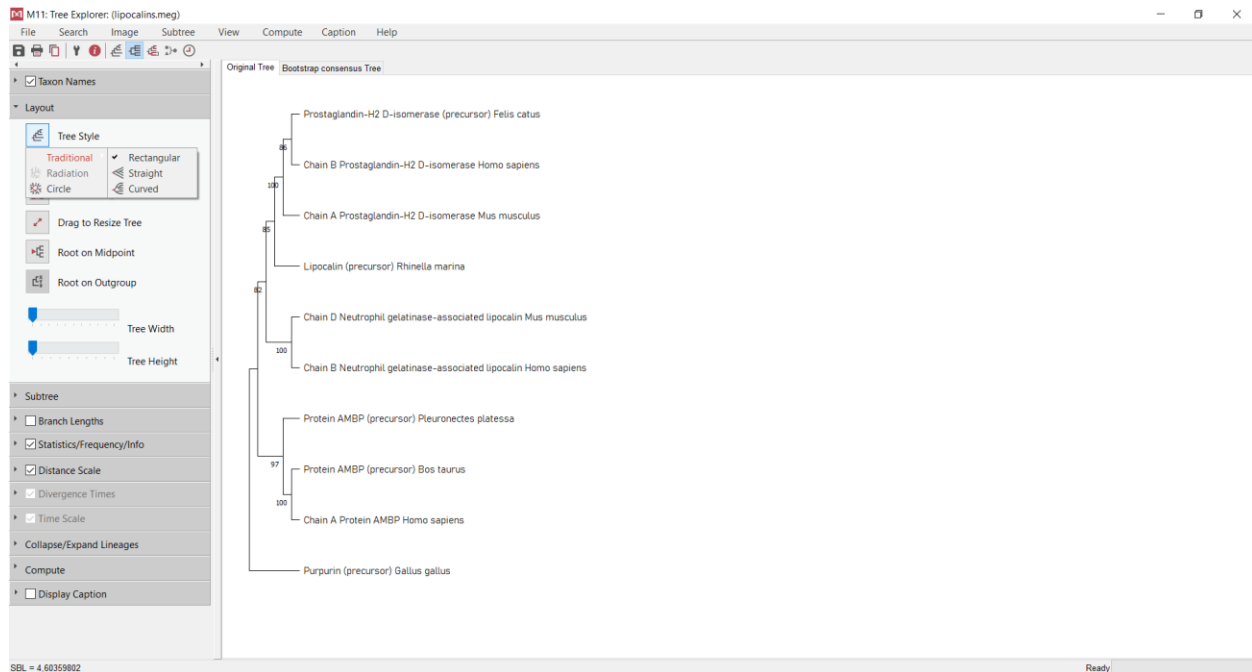
Το εργαλείο Taxon Names χρησιμοποιείται για την αλλαγή της γραμματοσειράς των ονομάτων των οργανισμών που εμφανίζονται στο δέντρο όπως αυτό που φαίνεται στο κόκκινο πλαίσιο



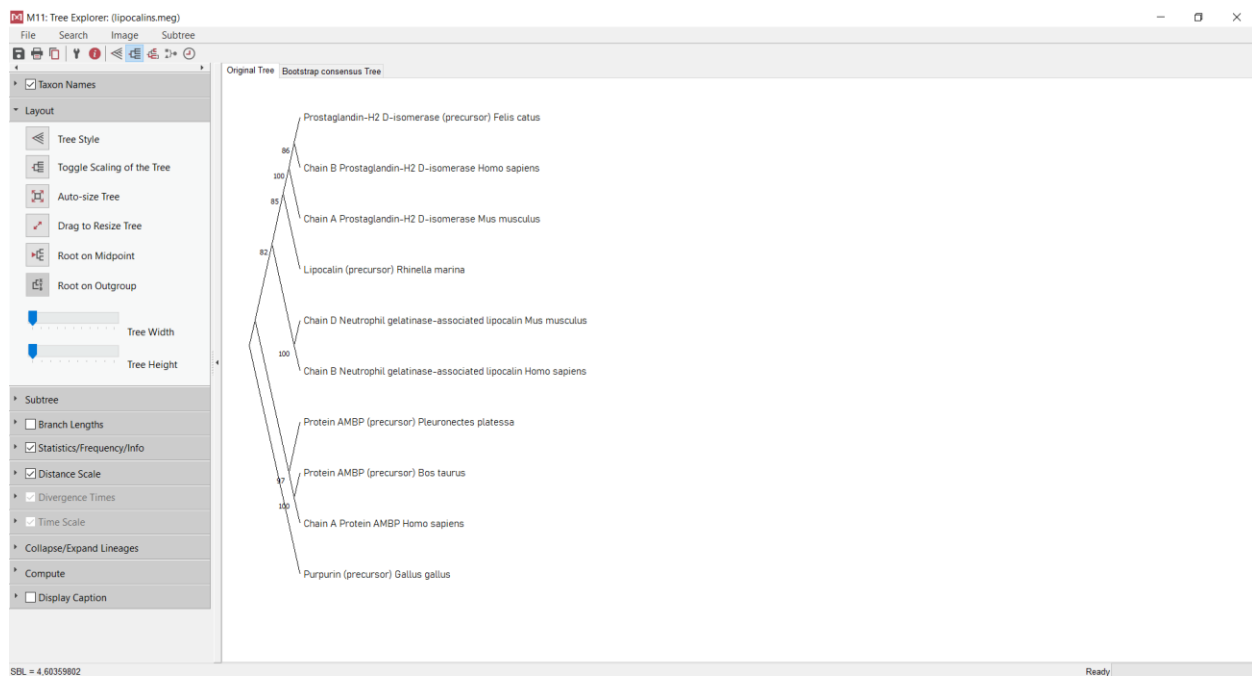
Το εργαλείο Layout αλλάζει τη μορφή απεικόνισης του προβαλλόμενου δέντρου



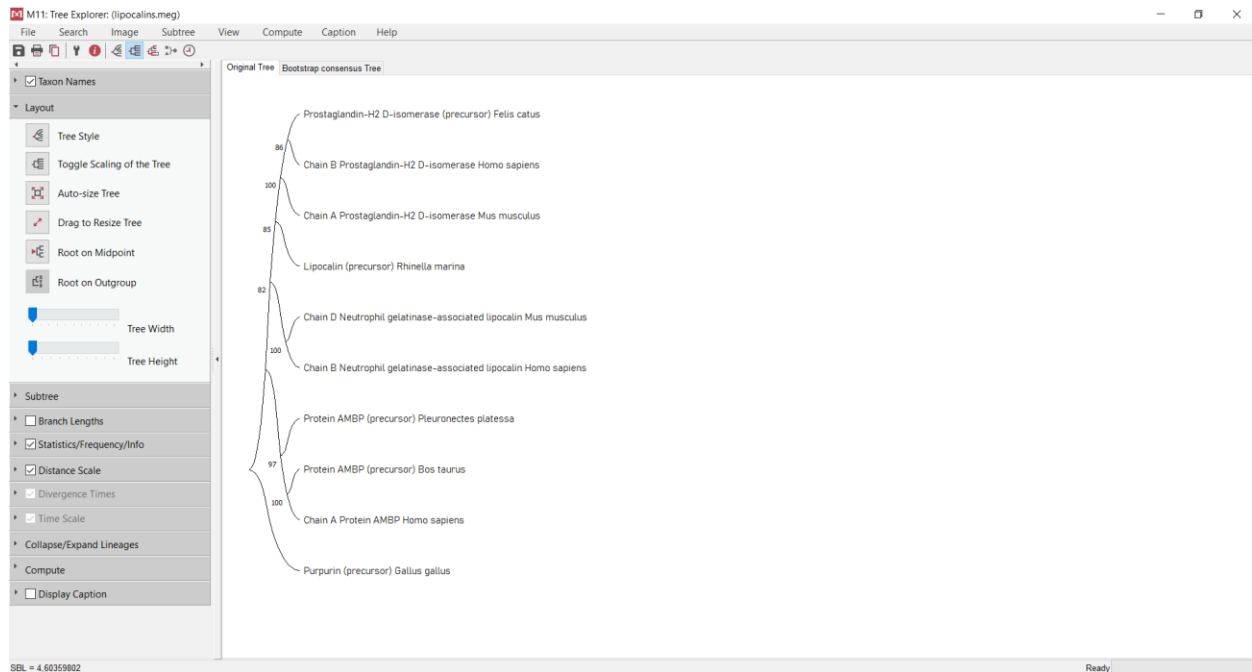
Αν πατήσουμε το Tree Style έχουμε διάφορες επιλογές όπως φαίνεται παρακάτω



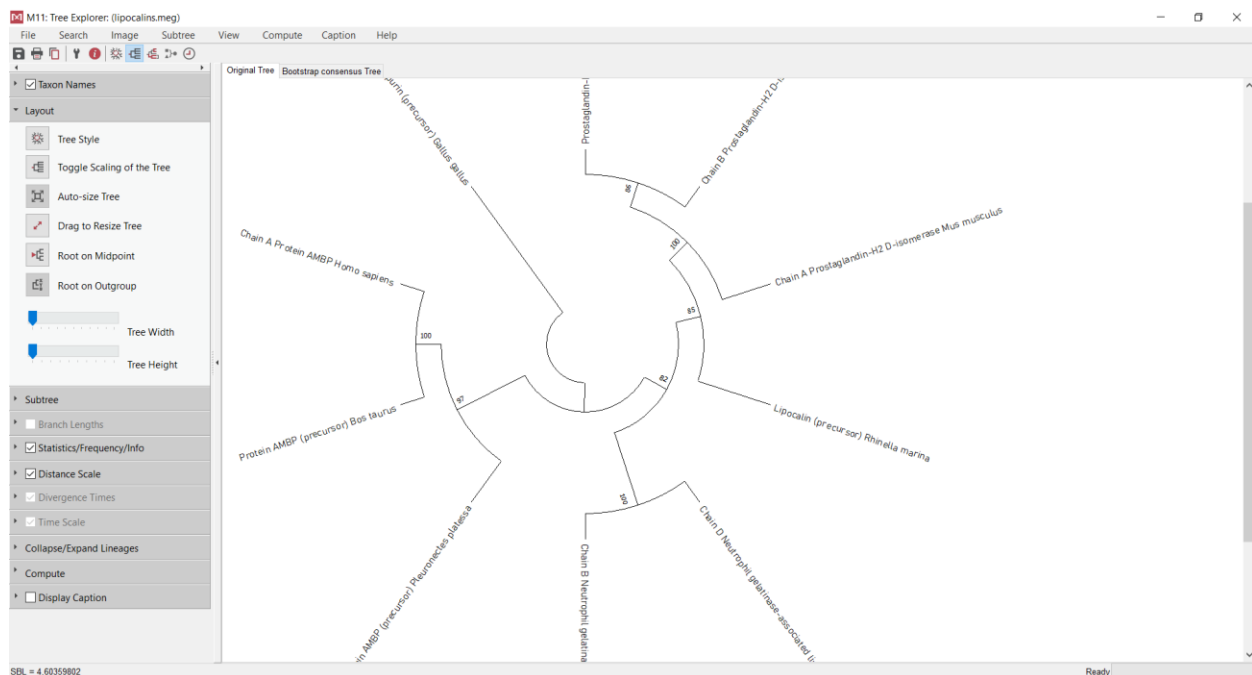
Αλλάζοντας την επιλογή από rectangular σε straight έχουμε το ακόλουθο αποτέλεσμα



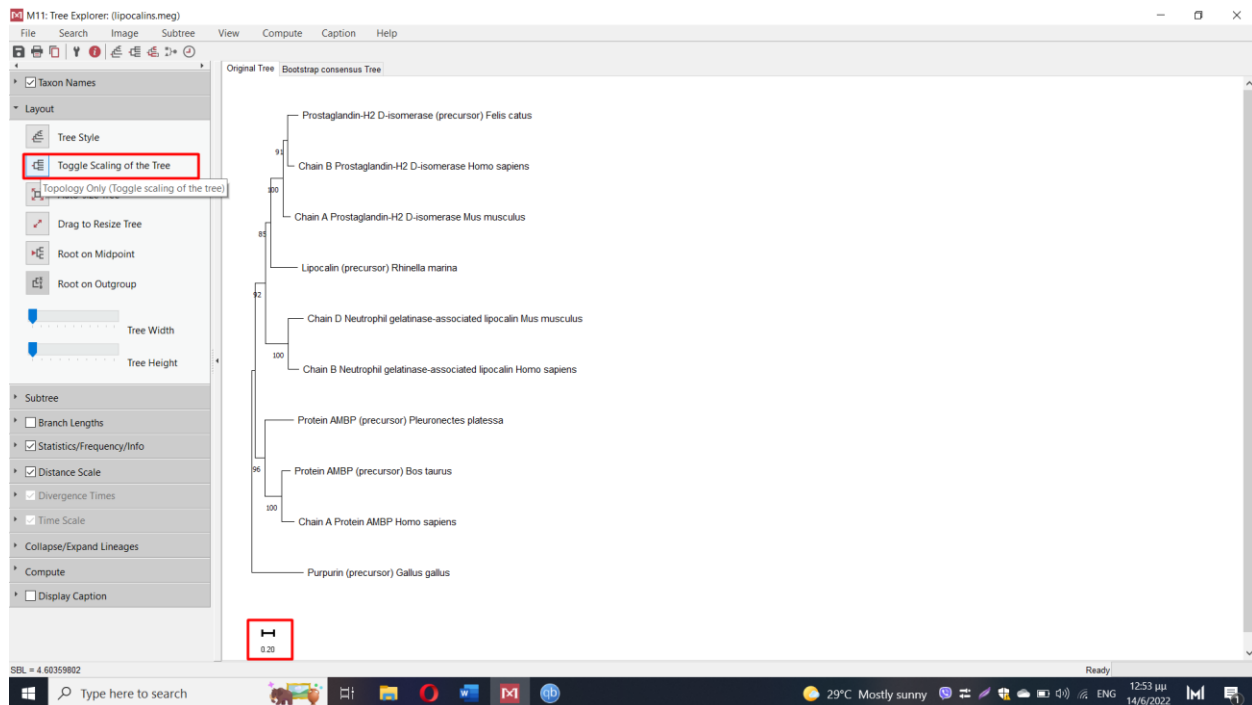
Ενώ αν επιλέξουμε curved το δέντρο εμφανίζεται ως εξής



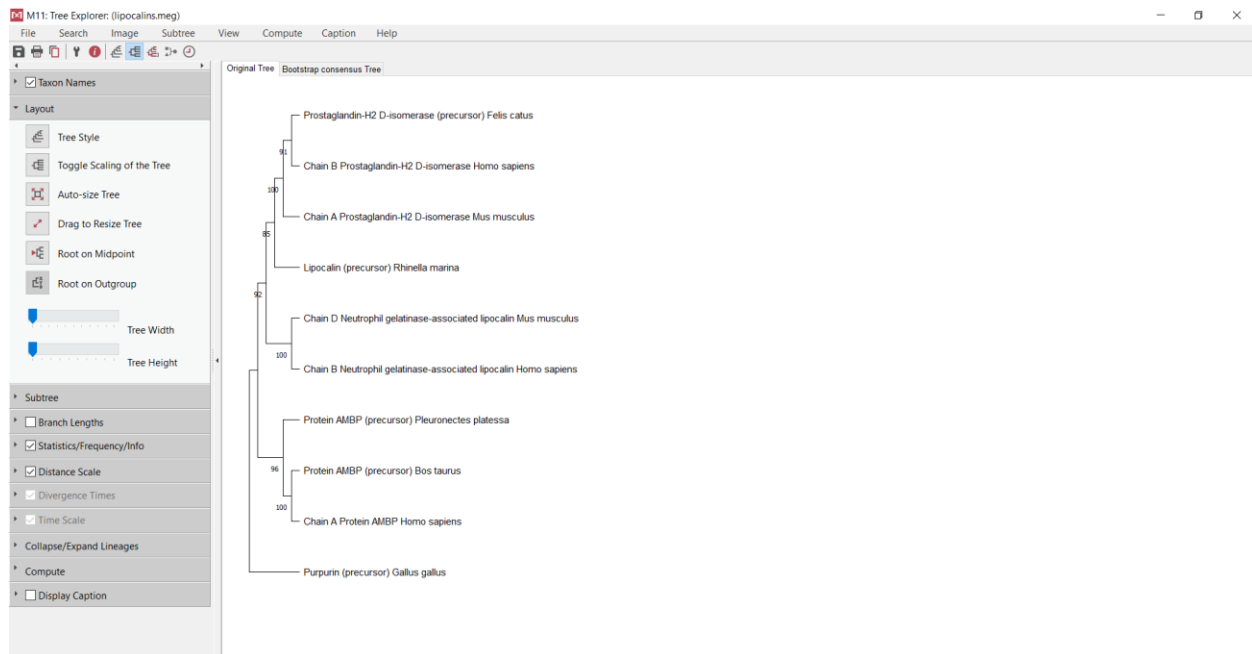
Αν αλλάξουμε τη μορφή απεικόνισης από traditional σε circled έχουμε τα ακόλουθα αποτελέσματα



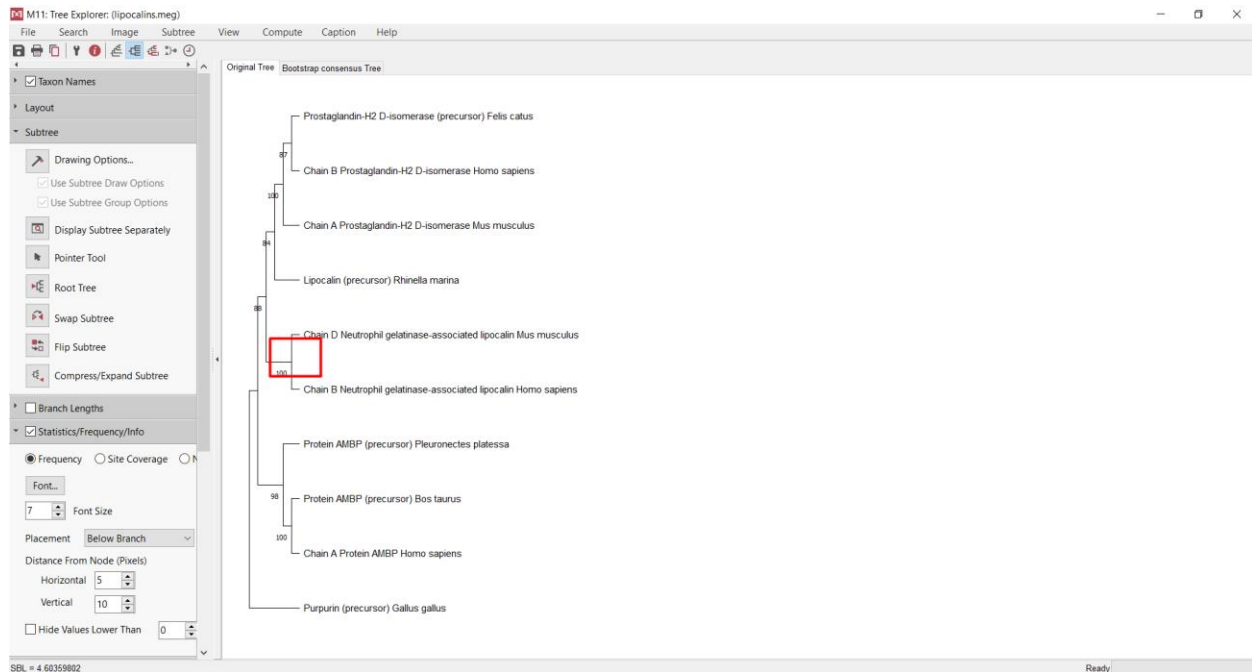
Στη συνέχεια έχουμε το εργαλείο Toggle Scaling of the Tree που μας επιτρέπει να εμφανίζουμε ή να εξαφανίζουμε την επιλογή εμφάνισης κλίμακας του δέντρου (Ο αριθμός 0.20 στο κάτω μέρος του σχήματος παρέχει μια κλίμακα για αυτό. Σε αυτήν την περίπτωση το ευθύγραμμο τμήμα με τον αριθμό 0.20 δείχνει το μήκος του κλάδου που αντιπροσωπεύει μια ποσότητα γενετικής αλλαγής 0.20)



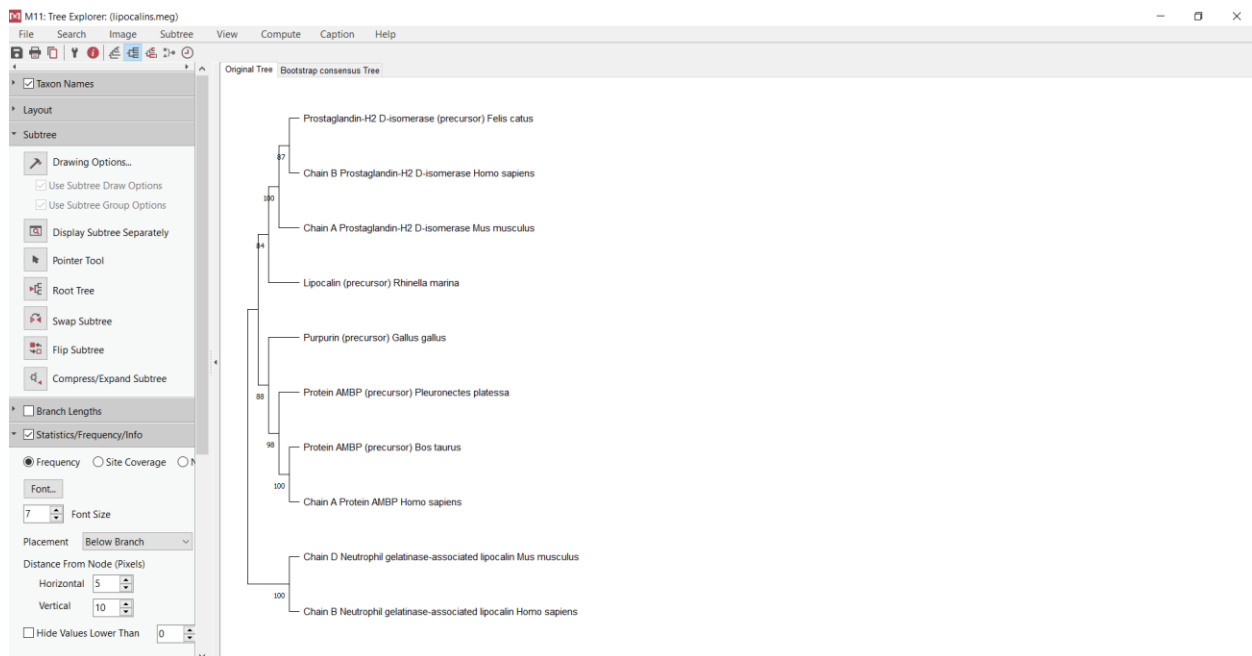
Αν πατήσουμε το κουμπί αυτό το σχήμα αλλάζει και γίνεται ως εξής



Τώρα με την επιλογή root tree θα δοκιμάσουμε να προσθέσουμε μια ρίζα σε κλαδιά που εμείς θα έχουμε επιλέξει. Για παράδειγμα αφού επιλέξουμε το root tree κάνουμε διπλό κλικ στο σημείο που βρίσκεται στο κόκκινο πλαίσιο



Και το δέντρο παίρνει την ακόλουθη μορφή



Γενικότερα το MEGA μας δίνει μια πληθώρα εργαλείων για να επεξεργαστούμε τόσο τη μορφή όσο και την απεικόνιση του δέντρου. Εκτός από τις λειτουργίες που εξετάσαμε αναλυτικά το MEGA μας επιτρέπει ακόμα να μεγεθύνουμε, συμπίεσουμε ή επεκτείνουμε ένα δέντρο, να αντιμετωπίσουμε δύο υποδέντρα, να «αναποδογυρίσουμε» τους κόμβους ενός υποδέντρου κ.α.

Βήμα 7: Πραγματοποίηση bootstrapping. Προσδιορισμός των συστάδων κλάδων που έχουν χαμηλά επίπεδα στήριξης. Γιατί συμβαίνει αυτό;

Κατά τη διαδικασία δημιουργίας φυλογενετικού δέντρου επιλέγουμε το Bootstrap Method

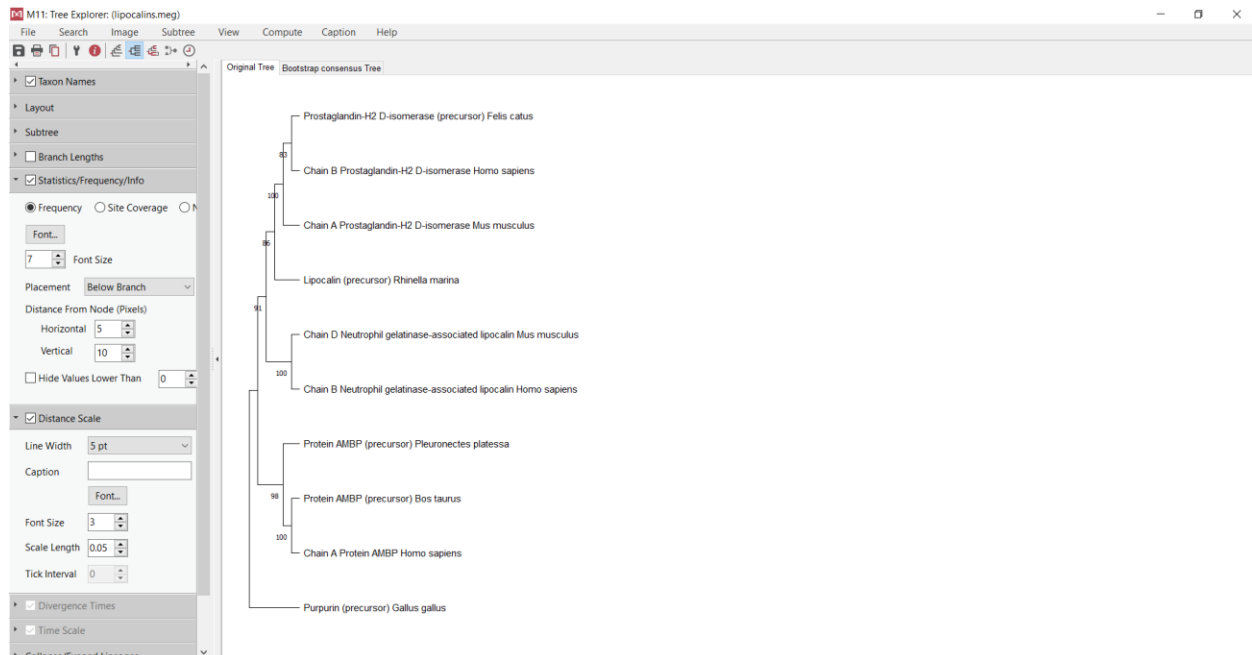
M11: Analysis Preferences

Phylogeny Reconstruction

Option	Setting
ANALYSIS	
Scope	→ All Selected Taxa
Statistical Method	→ Neighbor-joining
PHYLOGENY TEST	
Test of Phylogeny	→ Bootstrap method
No. of Bootstrap Replications	→ 100
SUBSTITUTION MODEL	
Substitutions Type	→ Amino acid
Model/Method	→ Poisson model
RATES AND PATTERNS	
Rates among Sites	→ Uniform Rates
Gamma Parameter	→ Not Applicable
Pattern among Lineages	→ Same (Homogeneous)
DATA SUBSET TO USE	
Gaps/Missing Data Treatment	→ Pairwise deletion
Site Coverage Cutoff (%)	→ Not Applicable
SYSTEM RESOURCE USAGE	
Number of Threads	→ 3

? Help X Cancel ✓ OK

Έχουμε έτσι το εξής δέντρο (είχαμε χρησιμοποιήσει από την αρχή το bootstrap method κατά τη δημιουργία του δέντρου οπότε θα είναι ίδιο με αυτό που έχουμε δείξει στο βήμα 6)

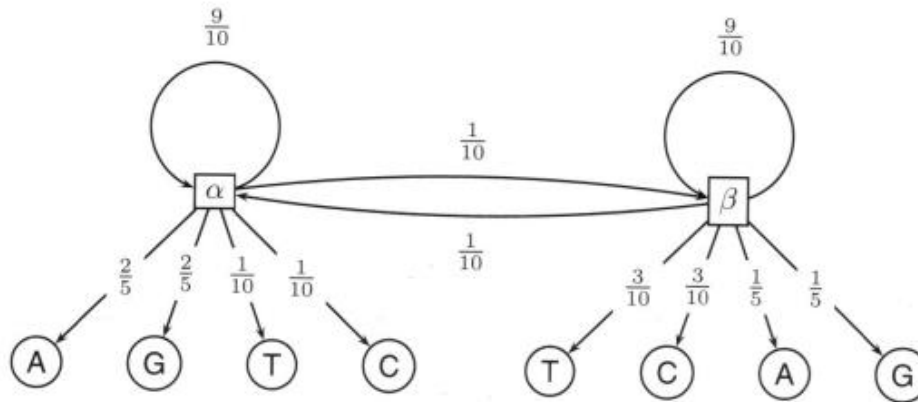


Παρατηρούμε ότι δίπλα από κάθε συστάδα κλάδου υπάρχει και ένας αριθμός. Στην περίπτωση μας τα νούμερα είναι από το 83 (το μικρότερο) έως και 100 (το μεγαλύτερο). Κάθε ένα από αυτά να νούμερα αντιπροσωπεύει τα επίπεδα στήριξης για την εκάστοτε συστάδα. Οι κόμβοι που συμμετέχουν σε συστάδες κλάδων με τα χαμηλότερα επίπεδα στήριξης είναι οι Prostaglandin-H2 D-isomerase (precursor) Felis catus και Chain B, Prostaglandin-H2 Disomerase Homo sapiens με τιμή 83. Η τιμή αυτή σημαίνει ότι η συστάδα κλάδων που περιλαμβάνει την Prostaglandin-H2 D-isomerase (precursor) του Felis catus και την Prostaglandin-H2 Disomerase του Homo Sapiens υποστηρίχθηκε στο 83% των δοκιμών το οποίο παρόλο ότι είναι χαμηλό νούμερο σε σχέση με τα υπόλοιπα δεν παύει να είναι ένα υψηλό ποσοστό. Γενικότερα η χρήση της μεθόδου bootstrapping χρησιμοποιεί την επαναλαμβανόμενη λήψη δειγμάτων ενός δοθέντος συνόλου δεδομένων για να παρέχει ένα μέτρο αξιολόγησης για το πόσο καλά μια τοπολογία δέντρου υποστηρίζεται από τα δεδομένα.

Θέμα 2^ο:

Ορισμός Προβλήματος:

Έχουμε στη διάθεση μας ένα Hidden Markov Model (HMM) το οποίο φαίνεται στην παρακάτω εικόνα



Καλούμαστε να αποκωδικοποιήσουμε την πιο πιθανή ακολουθία καταστάσεων (α/β) για την αλληλουχία GGCT (χρησιμοποιώντας λογαριθμικές βαθμολογίες αντί για κανονικές βαθμολογίες πιθανοτήτων). Στόχος είναι η υλοποίηση ενός προγράμματος σε **MATLAB** ή **PYTHON** το οποίο θα εφαρμόζει τον αλγόριθμο Viterbi που χρησιμοποιείται για την επίλυση τέτοιου είδους προβλημάτων.

Αλγοριθμική προσέγγιση:

Ο αλγόριθμος Viterbi είναι ο αλγόριθμος που πρέπει να υλοποιηθεί ώστε να λυθεί το συγκεκριμένο πρόβλημα και για αυτό χρησιμοποιείται η γλώσσα προγραμματισμού **PYTHON**. Ο αλγόριθμος Viterbi είναι ένας αλγόριθμος δυναμικού προγραμματισμού για τον υπολογισμό της μέγιστης εκτίμησης πιθανότητας της πιο πιθανής ακολουθίας κρυφών καταστάσεων - που ονομάζεται μονοπάτι Viterbi - που οδηγεί σε μια ακολουθία παρατηρούμενων γεγονότων, συγκεκριμένα στο πλαίσιο των πηγών πληροφοριών Markov και Hidden Markov Model (HMM). Ο αλγόριθμος θα είναι μια συνάρτηση **PYTHON** ο οποίος θα δέχεται ως είσοδο:

- Το χώρο των καταστάσεων (στην περίπτωση μας το α και το β)
- Το χώρο των παρατηρήσεων (GGCT στην περίπτωση μας)
- Τις πιθανότητες μετάβασης ($9/10$ και $1/10$ στην περίπτωση μας)
- Τις πιθανότητες εκπομπής ($2/5, 1/10, 3/10, 1/5$ στην περίπτωση μας)
- Τις πιθανότητες του κόμβου έναρξης (στην περίπτωση μας δεν καθορίζονται)

Και θα παράγει ως έξοδο μια ακολουθία από τις δοσμένες καταστάσεις (α, β) καθώς και έναν αριθμό που θα αναπαριστά τη μέγιστη εκτίμηση πιθανότητας για την συγκεκριμένη ακολουθία.

Όπως ξέρουμε ο αλγόριθμος Viterbi -για το σύνολο των καταστάσεων που υπάρχουν- υπολογίζει σε κάθε μετάβαση, για κάθε έναν από τους κόμβους της δοσμένης αλληλουχίας, την εκτίμηση πιθανότητας μιας συγκεκριμένης ακολουθίας καταστάσεων πολλαπλασιάζοντας την πιθανότητα μετάβασης από την τρέχουσα κατάσταση με την προηγούμενως υπολογισμένη εκτίμηση πιθανότητας για τον προηγούμενο κόμβο στη δοσμένη αλληλουχία, και την πιθανότητα εκπομπής για τον τρέχον κόμβο της αλληλουχίας από την τρέχουσα κατάσταση.

Λεπτομέρειες υλοποίησης:

Το πρόγραμμα που αναπτύξαμε για την εξαγωγή της πιο πιθανής ακολουθίας καταστάσεων για μια δοσμένη αλληλουχία νουκλεοτιδίων είναι σε **PYTHON**. Για την αποθήκευση των υπολογισμένων από το πρόγραμμα δεδομένων χρησιμοποιούμε **Dictionaries** και για την αναπαράσταση των δεδομένων εισόδου χρησιμοποιούμε κάποια μονοδιάστατα και δισδιάστατα **arrays**. Έχουμε δημιουργήσει συνολικά δύο συναρτήσεις, η μια εξυπηρετεί την εισαγωγή στοιχείων σε ένα dictionary σε ένα συγκεκριμένο key ενώ η άλλη είναι η υλοποίηση του αλγορίθμου **Viterbi**. Δέχεται 5 ορίσματα (οι είσοδοι που αναφέραμε παραπάνω με τις πιθανότητες του κόμβου έναρξης να είναι προαιρετικές). Σε περίπτωση που η προαιρετική είσοδος δεν δοθεί το πρόγραμμα θεωρεί ισοπίθανη την έναρξη για κάθε μια από τις υπάρχουσες καταστάσεις. Υπολογίζει εκτός επανάληψης τις δύο πρώτες τιμές εκτίμησης πιθανότητας για τις δύο καταστάσεις και ύστερα μέσα σε μια επανάληψη υπολογίζει την εκτίμηση πιθανότητας από κάθε κατάσταση για κάθε νουκλεοτίδιο της αλληλουχίας. Στην περίπτωση διακλάδωσης ελέγχουμε τις τιμές εκτίμησης πιθανότητας των δύο κλάδων που προκύπτουν και εφόσον βρισκόμαστε σε σημείο που είναι αποδεκτό από τον αλγόριθμο απορρίπτουμε την χαμηλότερη τιμή. Παράλληλα υπολογίζουμε με τη βοήθεια ενός δεύτερου **dictionary** το μονοπάτι που οδηγεί στην εκάστοτε τιμή εκτίμησης πιθανότητας έτσι ώστε να το εμφανίσουμε στο τέλος. Όταν βεβαιωθούμε ότι έχουμε περάσει από όλα τα νουκλεοτίδια της αλληλουχίας ο αλγόριθμος ολοκληρώνεται. Υπολογίζουμε την μέγιστη τιμή από το **dictionary** που περιέχει τα **Viterbi paths** και με βάση αυτή τη τιμή εξάγουμε και την αλληλουχία των καταστάσεων. Εκτυπώνουμε και τις δυο αυτές τιμές στην οθόνη. Όταν καλούμε τη συνάρτηση βάζουμε τις απαραίτητες τιμές στα ορίσματα και με τη βοήθεια της βιβλιοθήκης **numpy** και της συνάρτησης **log** εισάγουμε τις λογαριθμικές βαθμολογίες πιθανοτήτων όπως ζητείται από την εκφώνηση.

Σημείωση: Αναλυτικότερες πληροφορίες σχετικά με την υλοποίηση του προγράμματος υπάρχουν στα σχόλια του κώδικα.

Έλεγχος εγκυρότητας προγράμματα μέσα από την εκτέλεση του κώδικα (αρχείο viterbi.py) :

```
C:\Users\evr\PycharmProjects\viterbi_algorithm\venv\Scripts\python.exe "C:\Users\evr\Desktop\UNIFI\3o έτος\6o εξάμηνο\Βιοπληροφορική\Απαλλακτική
εργασία\viterbi_algorithm\viterbi.py"
highest probability -5.442982980493551
Sequence ['b', 'b', 'b', 'b']

Process finished with exit code 0
```

Μπορείτε να δείτε αναλυτικά όλα τα βήματα του αλγορίθμου βγάζοντας από τα σχόλια τις εντολές **print()**.

Θέμα 3ο:

Ορισμός Προβλήματος:

Επιλέχθηκε η υλοποίηση του προβλήματος **6.12** του βιβλίου "Εισαγωγή στους Αλγορίθμους Βιοπληροφορικής". Σύμφωνα με την εκφώνηση δύο παίκτες παίζουν το εξής παιχνίδι με δυο χρωμοσώματα που έχουν μήκος n και m νουκλεοτίδια αντίστοιχα. Σε κάθε γύρο του παιχνιδιού, ένας παίκτης μπορεί να καταστρέψει ένα από τα χρωμοσώματα και να διαχωρίσει το άλλο σε δύο μη κενά τμήματα. Ο παίκτης που διαγράφει το τελευταίο νουκλεοτίδιο κερδίζει. Στόχος είναι η υλοποίηση ενός προγράμματος σε **MATLAB** ή **PYTHON** το οποίο θα περιγράφει τη νικηφόρα στρατηγική για οποιαδήποτε n και m νουκλεοτίδια.

Αλγοριθμική προσέγγιση:

Το συγκεκριμένο παιχνίδι έχει κάποιες καταστάσεις στις οποίες αν βρεθεί ο παίκτης αυξάνονται κατά πολύ οι πιθανότητες νίκης και αντίστοιχα κάποιες άλλες στις οποίες αυξάνονται οι πιθανότητες ήττας. Η βέλτιστη στρατηγική είναι να οδηγούμε με κάθε κίνηση μας τον εαυτό μας σε κατάσταση πιθανής νίκης ενώ τον αντίπαλο σε κατάσταση πιθανής ήττας. Σε κάθε κίνηση ο παίκτης επιτρέπεται να καταστρέψει ένα χρωμόσωμα και να διαχωρίσει ένα άλλο. Είναι εύκολο να κατανοήσουμε πως η μια θέση όπου ο αντίπαλος έχει στη διάθεση του μονά νουκλεοτίδια οδηγεί σε κατάσταση πιθανής ήττας τον αντίπαλο αφού οι μονοί αριθμοί μπορούν να σπάσουν σε έναν μονό και ένα ζυγό που είναι αντίστοιχα θέση νίκης για εμάς αφού εμείς αντίστοιχα μπορούμε να συνεχίσουμε να σπάμε τους ζυγούς σε δύο μονούς οδηγώντας τον αντίπαλό σταδιακά στο να στο να ξεμείνει από κινήσεις αφού όλες οι νικητήριες θέσεις είναι ζυγές. Επομένως μέρος της βέλτιστης στρατηγικής είναι να **σπάμε τα ζυγά νουκλεοτίδια -όταν αυτά υπάρχουν- σε δύο μονά**. Ταυτόχρονα πρέπει να αποφύγουμε να οδηγήσουμε τον παίκτη σε θέση όπου έχει στην διάθεση του χρωμοσώματα με μήκος **2** ή **4** καθώς θα οδηγηθούμε πιθανότατα σε ήττα και να επιδιώξουμε να τον οδηγήσουμε σε θέσεις **3** ή **5** καθώς θα οδηγηθούμε πιθανότατα σε νίκη. Συνδυάζοντας αυτούς τους περιορισμούς παίρνουμε τη βέλτιστη στρατηγική η οποία αν εφαρμοστεί είναι ικανή να δώσει σε έναν παίκτη τη νίκη σε ένα πολύ μεγάλο ποσοστό των περιπτώσεων αν όχι σε όλες.

Λεπτομέρειες υλοποίησης:

Το πρόγραμμα που αναπτύξαμε για να προσομοιώσουμε την βέλτιστη στρατηγική σε έναν γύρο παιχνιδιού είναι σε γλώσσα προγραμματισμού **PYTHON**. Κατά την εκτέλεση του έχουμε δύο παίκτες: τον άνθρωπο που κάνει κινήσεις με τυχαία στρατηγική με τη βοήθεια της βιβλιοθήκης **random** και τον υπολογιστή που κάνει κινήσεις με βάση την βέλτιστη στρατηγική που περιεγράφηκε πιο πάνω. Οι παίκτες παίζουν εναλλάξ, πραγματοποιούν τις κινήσεις και όταν το παιχνίδι τελειώσει υπάρχει ένας νικητής. Το πρόγραμμα πραγματοποιεί 100 γύρους και στο τέλος εκτυπώνει πόσους από αυτούς κέρδισε ο υπολογιστής και πόσους από αυτούς κέρδισε ο παίκτης. Το πρόγραμμα αρχικοποιείται δημιουργώντας τα χρωμοσώματα n και m . Παίρνουμε τις αλληλουχίες από το προτεινόμενο σύνδεσμο και τις επικολλάμε σε ένα αρχείο `fn`. Φτιάχνουμε δύο αρχεία -ένα για το κάθε χρωμόσωμα- τα οποία μέσω της συνάρτησης **open()** της **PYTHON** τα διαβάζουμε μέσα σε δύο μεταβλητές. Στη συνέχεια ορίζουμε σε μια μεταβλητή μια λογική τιμή (`true` ή `false`) που ορίζει ποιος παίκτης θα ξεκινήσει το παιχνίδι. Ο παίκτης στη σειρά του κάνει μια τυχαία κίνηση με την εξής διαδικασία:

- Επιλέγεται ένας τυχαίος ακέραιος.
- Ανάλογα με αυτόν τον αριθμό ο παίκτης επιλέγει πιο χρωμόσωμα θα καταστρέψει και πιο θα διαχωρίσει.
- Επιλέγεται ένας τυχαίος αριθμός που καθορίζει το σημείο του χρωμοσώματος στο οποίο θα γίνει ο διαχωρισμός
- Δημιουργούνται δύο νέα χρωμοσώματα που είναι αποτελέσματα του διαχωρισμού του επιλεγμένου χρωμοσώματος

Βέβαια υπάρχουν κάποιες κινήσεις που πρέπει να γίνονται ανεξάρτητα από την τυχαιότητα για να εξασφαλιστεί η εγκυρότητα του παιχνιδιού. Για παράδειγμα αν ένα χρωμόσωμα έχει μήκος 1 προφανώς δεν μπορεί να διασπαστεί σε δύο μέρη σύμφωνα με τους κανόνες του παιχνιδιού οπότε θα είναι υποχρεωτικά το χρωμόσωμα που θα πρέπει να καταστραφεί. Επομένως πριν από κάθε τυχαία κίνηση του παίκτη ελέγχουμε αν υπάρχουν τέτοιου είδους περιπτώσεις ώστε η κίνηση του παίκτη να είναι σίγουρα έγκυρη. Όταν ολοκληρωθεί η σειρά του παίκτη ακολουθεί η κίνηση του υπολογιστή. Ο υπολογιστής παίζει με την στρατηγική που αναφέραμε παραπάνω δηλαδή:

- Ελέγχει αν υπάρχει κάποιο ζυγού μήκους χρωμόσωμα
- Εάν υπάρχει τότε διαχωρίζει το ζυγό σε δύο μονά και καταστρέφει το άλλο
- Εάν δεν υπάρχει ζυγό τότε διαλέγει ένα από τα μονά (Εάν υπάρχει στη διάθεση του υπολογιστή μια από τις επιθυμητές θέσεις (3 ή 5) τότε επιλέγει αυτή)
- Επιλέγεται ένας τυχαίος αριθμός που καθορίζει το σημείο του χρωμοσώματος στο οποίο θα γίνει ο διαχωρισμός
- Εξασφαλίζεται ώστε ο τυχαίος αριθμός να μην οδηγεί σε κάποιο μη επιθυμητό διαχωρισμό (χρωμόσωμα μήκους 2 ή 4) εκτός και αν δεν υπάρχει άλλη πιθανή κίνηση

Όταν ολοκληρωθεί μια παρτίδα (μια παρτίδα ολοκληρώνεται όταν έχουν μείνει δυο χρωμοσώματα με μήκος ≤ 1) τότε προστίθεται σε έναν μετρητή μια μονάδα που κρατάει το σκορ νικών του παίκτη και του υπολογιστή αντίστοιχα. Αν ο παίκτης βρεθεί στη διάθεση του με δύο χρωμοσώματα μήκους 1 τότε με βάση τους κανόνες του παιχνιδιού θεωρείται ότι ο υπολογιστής νικά και το αντίθετο γίνεται αν ο υπολογιστής βρεθεί στη διάθεση του με δύο χρωμοσώματα μήκους 1.

Σημείωση: Αναλυτικότερες πληροφορίες σχετικά με την υλοποίηση του προγράμματος υπάρχουν στα σχόλια του κώδικα.

Έλεγχος εγκυρότητας προγράμματα μέσα από την εκτέλεση του κώδικα (αρχείο `ex3\main.py`):

```
C:\Users\evriv\PycharmProjects\ex_3\venv\Scripts\python.exe C:/Users/evriv/PycharmProjects/ex_3/main.py
Iterations where the computer won
100
Iterations where the player won
0
Process finished with exit code 0
```

Παρατηρούμε ότι από τα 100 παιχνίδια ο υπολογιστής που ακολουθούσε την προτεινόμενη στρατηγική κέρδισε και τα 100 ενώ ο παίκτης δεν κέρδισε κανένα. Μπορείτε να δείτε αναλυτικά τις κινήσεις τόσο του παίκτη όσο και του υπολογιστή βγάζοντας από τα σχόλια τις εντολές **print()**.

Θέμα 4^ο

Μας ζητήθηκε να αναζητήσουμε τον κωδικό **PDB-code: 7NEH** στην πρωτεϊνική βάση δεδομένων <https://www.rcsb.org/> και με την χρήση του προγράμματος **Chimera-X** να απαντήσουμε τα παρακάτω ερωτήματα.

Ερώτημα 1:

- a) Δείτε τα στοιχεία που παρουσιάζονται στην πρωτεϊνική βάση δεδομένων και προσδιορίστε τη μέθοδο με την οποία έχει προσδιορισθεί η δομή του συμπλόκου;
Απάντηση: Η μέθοδος που χρησιμοποιήθηκε για να προσδιορισθεί η δομή του συμπλόκου ονομάζεται: **X-RAY DIFFRACTION**
- b) Ποιο το resolution (διακριτική ικανότητα) στο οποίο προσδιορίστηκε η δομή;
Απάντηση: Το resolution στο οποίο προσδιορίστηκε η δομή είναι **1.77 Å**.
- c) Παραθέστε το Ψηφιακό αναγνωριστικό (Digital Object Identifier, DOI) της σχετικής επιστημονικής δημοσίευσης.
Απάντηση: PDB DOI: 10.2210/pdb7NEH/pdb

Ερώτημα 2:

- a) Πόσες διακριτές πρωτεϊνικές αλυσίδες (molecular entities, macromolecules) περιλαμβάνει η εν λόγω δομή;
Απάντηση:
 - 1. COVOX-269 Fab heavy chain
 - 2. COVOX-269 fab light chain
 - 3. Spike glycoprotein
- b) Για κάθε μια από αυτές σημειώστε το πλήθος των αμινοξέων (sequence length)
Απάντηση:
 - 1. COVOX-269 Fab heavy chain πλήθος: 222
 - 2. COVOX-269 fab light chain πλήθος: 215
 - 3. Spike glycoprotein πλήθος: 205
- c) Πόσους ολιγοσακχαρίτες περιλαμβάνει η δομή του συμπλόκου;

Απάντηση:

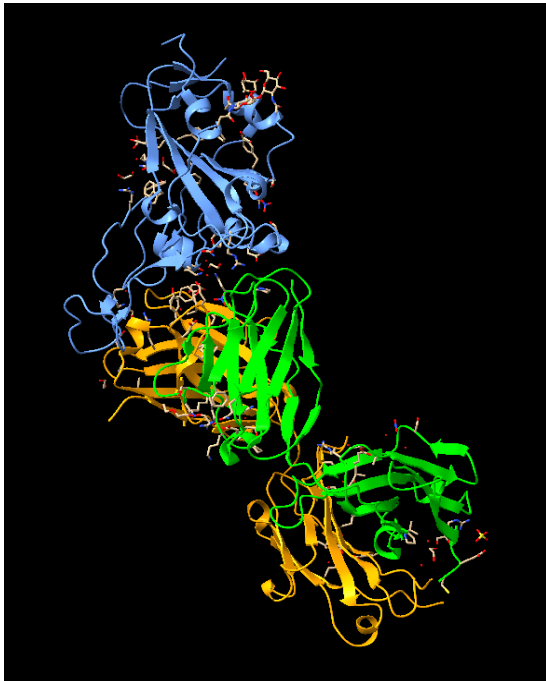
2-acetamido-2-deoxy-beta-D-glucopyranose-(1-4)-[alpha-L-fucopyranose-(1-6)]2-acetamido-2-deoxy-beta-D-glucopyranose

d) Η δομή του συμπλόκου έχει ένα άτομο χλωρίου (Cl⁻). Παραθέστε την αλυσίδα την οποία ανήκει

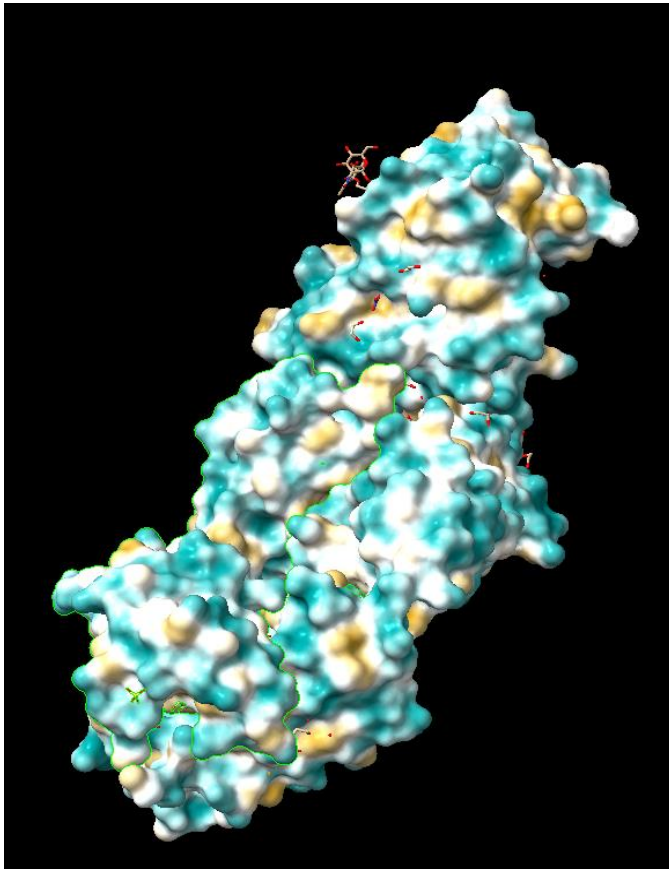
Απάντηση:

Ανήκει στην αλυσίδα **CA** [auth E] .

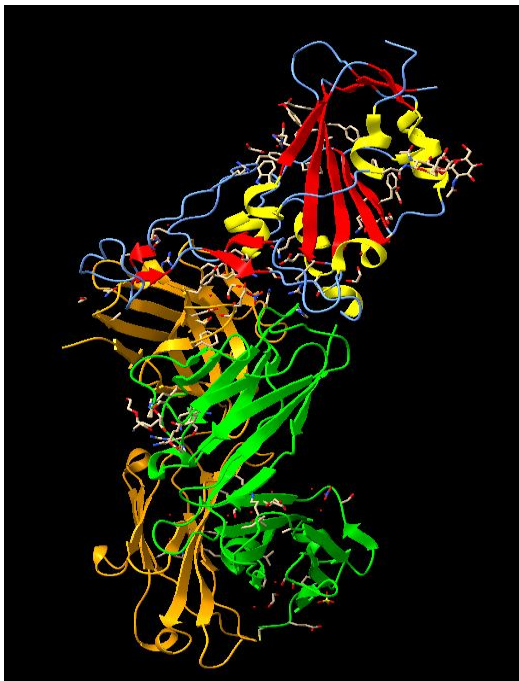
Ερώτημα 3:



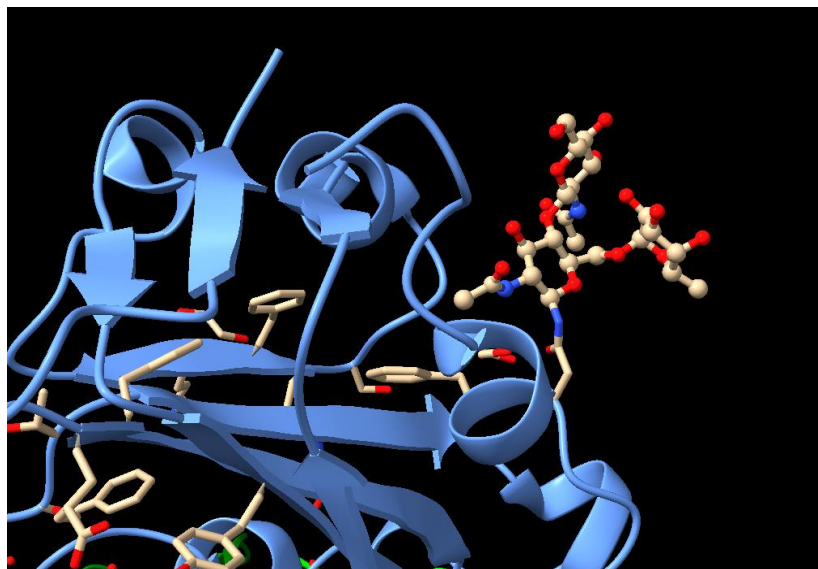
Ερώτημα 4:



Ερώτημα 5: A)



B)



Г)

