Practical Computing for Bioinformatics
Assingment I

Alexandra Pančíková
Ivana Janíčková

November 2, 2020

Exercise 1

1.a

NCBI-Protein. NCBI search of the given IDs was redirected to the NCBI-protein database.

1.b

Corresponding gene name was found in the NCBI Protein database GenPept format in the CDS section. Corresponding organism can be seen in the definition of the protein, or alternatively by directly searching the gene NCBI Gene database.

- Gene: CDC42BPA
- Gene Synonym: MRCK, MRCKA, PK428
- Organism: Homo Sapiens

1.c

Homolog in *C. elegans* is **MRCK-1**
Homologous genes in *C. elegans* was found by querying NCBI HomoloGene database with the human gene name. The query output listed the homolog genes found in various organisms. The *C. elegans* homolog MRCK-1 was found.
Gene knockdown - The WormBase database contains information about phenotype, where we can select evidence based on RNAi experimental data. Knockdown causes slow growth phenotype.[1] During embryonic development, the activity of MRCK-1 is necessary for apicobasal polarization of cadherin-GFP localization. Knock down of MRCK-1 disrupts the apicobasal polarization of HMR-1/cadherin-GFP, hence weakening the apical junction between the endoderm precursor cells (Ea and Ep).[2]

1.d

One of the ways for retrieving a list of GO terms for a given gene sequence is by clicking on Complete GO annotation on QuickGO at the Uniprot page

of the observed gene. The link is redirected to the EBI QuickGO page where all the associated GO terms are listed. These can be exported in various formats - we chose TSV. We can compare the common GO terms by Unix command `comm -1 -2`, which returns GO terms common to both sequences. We then save the common GO terms to the txt file.
Examples of common GO terms:

- GO:0004674
- GO:0000166
- GO:0106310

Exercise 2

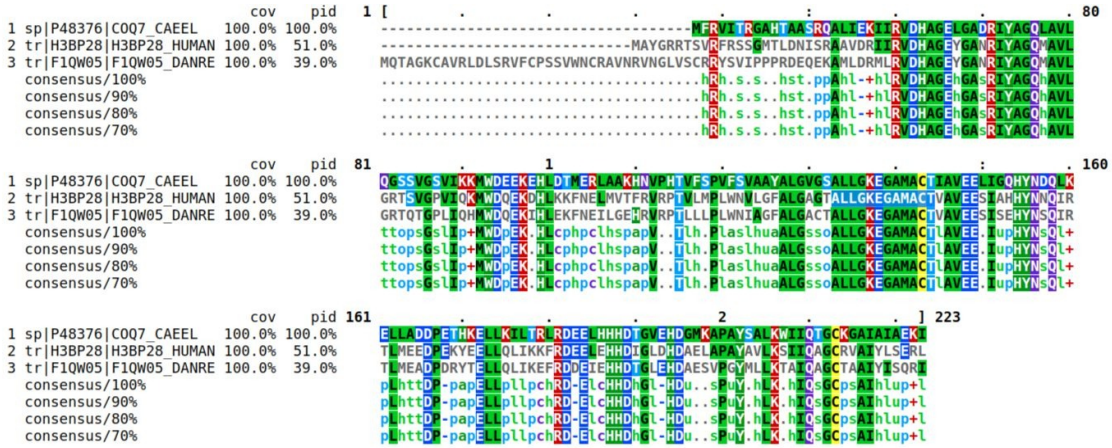
2.a

In order to find the species to which the unknown sequence belongs, we use the NCBI's BLAST tool to which we enter the sequence. Blast will search for matching sequences in organisms.

- The resulting organism is: *Caenorhabditis elegans*.
- The corresponding gene is **clk-1** and codes for 5- demethoxyubiquinone hodroxylase.

2.b

We find the homologs to our gene by browsing Ensembl with the name and organism of the observed gene. Then, in the comparative genomics section we go to orthologues and find the orthologous genes in H. Sapiens and D. Rerio. We download the corresponding protein sequences in FASTA format from UniProt website.
To Perform a multiple sequence alignment we run the msa - Clustal Omega tool on the EBI page. For visualization we enter the output to MView. Below is the picture of the visualization. Matched regions of the sequences are highlighted. Below the four consensus rows are computed at stated thresholds of percentage composition of the columns. The cov column describes the percentage coverage of every seq. with respect to reference - which is in our



case the *C. elegans* sequence. It is calculated as *(the number of residues aligned with reference) / (the length of reference row) x 100*. In the case of our alignments it is equal to 100% in all alignments, meaning that the number of residues aligned with reference equals the length of the reference row. The pid column describes a percent identity calculated as *(the number of identical residues)/ (the length of reference row) x 100*. The alignment of *H. sapien* has therefore 51% of identical residuals with respect to C elegans, and the alignment of *D. rerio* is 39% identical to the reference. The values are relatively small, however we may observe a conserved region (132 -144), suggesting the importance of this sequence.

The percentage of homology with respect to reference sequence - *C. elegans*:

H. sapiens = 50%

D. rerio = 39%

Exercise 3

3.a

First we list all the files from the log_files directory. By searching for 'stderr' we filter out error files from jobs run on the VSC cluster. Command wc -l counts the number of files that report on the errors.

Number of error files = 529

```
ls | grep 'stderr' | wc -l
```

In order to list all error job IDs we first create an empty file to which we will later list found job IDs.
`touch error_jobID.txt`
Using grep -i command we find string that contains error/Error string, which excludes empty files. We filter for 'stderr' files. To obtain a list containing only job IDs, we cut the output into columns by using "." as a delimiter and then we choose the second column which contains the job IDs. Next we filter out duplicate IDs by unique command. We then redirect the output of command to the later created error_jobID.txt file.
`grep -li error stderr* | cut -f2 -d"." | uniq > error_jobID.txt`

3.b

Finally ,we count the number of job IDs using `wc -l` command, which counts the number of lines in the job_ID.txt file - where each line corresponds to one unique job ID.
`wc -l error_jobID.txt`
The number of jobIDs = 86

Exercise 4

Please see the commented code in the `cputime.sh` file in log_files directory.

Bibliography

[1] Simmer F, Moorman C, van der Linden AM, Kuijk E, van den Berghe PV, et al. (2003) Genome-Wide RNAi of C. elegans Using the Hypersensitive rrf-3 Strain Reveals Novel Gene Functions. PLOS Biology 1(1): e12.

[2] Marston DJ, Higgins CD, Peters KA, et al. MRCK-1 Drives Apical Constriction in C. elegans by Linking Developmental Patterning to Force Generation. Curr Biol. 2016;26(16):. doi:10.1016/j.cub.2016.06.010