In the following exercise we'll learn how to search for sequences in different sequence databases. Examples for using sequence search are:

- To check whether your sequence is new or if it already exists in the database.
- To find homolog sequences.
- To find common motifs

We will use the BLAST package which is probably the most well-known tool in Bioinformatics

Let's enter the NCBI blast main page using the following link:

https://blast.ncbi.nlm.nih.gov/Blast.cgi

Various programs are presented at this page. There are two key differences between each of them: the type of sequence or sequences that we use as input and which database we use for the search. The following programs are available on NCBI:

1**. Blast Genomes**- used to run a sequence against a selected genome.

2. **Nucleotide Blast- blastn**- used to run a nucleotide sequence against the nucleotide database.

3. **Protein BLAST- blastp**- used to run a protein sequence against the protein database.

4. **Blastx**- used to run a translated nucleotide sequence against the protein database.

5. **Tblastn**- used to run a protein sequence against the translated nucleotide database

6. **Tblastx**- used to run a translated nucleotide sequence against the translated nucleotide database – This option is not available from the blast main page, but can be reached via each one of the blast running program pages at the upper tabs.

Please note that in general if we have both the nucleotide and the protein sequences of the same gene, it is better to perform a protein sequence search than a nucleotide sequence search, as the former is usually more sensitive.

There are also specialized search options that appear at the bottom of the page. For example: primer design, global alignment, Immunoglobulin and T-cell receptor search and more.

Let's run the Blastp program (using default values) on the following unknown protein sequence.

```
>unknown protein
AAAQHLCGSHLVDALYLVCGEKGFFYNPKGIVEQCCHKPCNIFDLQNYCN
```

Open the blastp input page and enter the sequence to the input window. Please note that you can define the sequence database or a specific organism against which the sequence will be run. Click on Blast to run the program.

The output page contains five sections.

The first part includes details on the running parameters. We can see that the NR database was used. The NR database includes non-redundant sequences from genbank, PDB, Swissprot and more. Clicking on the “search summary” tab will reveal that the NR database includes about 195 million sequences against which our sequence was run.

The second part includes a plot of our sequence, with markings of the conserved domains that were found by the search.

The third part includes a graphical view of the results. Each line represents one result and the line length is proportional to the sequence coverage. The line color is based on a blast score, with high scores shown as red or pink. Please note that it doesn’t mean that other colors are not meaningful. In general, we don’t take into account the blast score parameter itself in order to filter the results, as there are other, more relevant, output parameters that will be mentioned below.

The fourth part includes the results table. Each line represents one sequence result with related output columns’ parameters as a percentage of identity. There are two main output parameters:

The first one is the “Query coverage”. That represents the percentage of the query sequence that’s aligned. For example: A result with 90% Similarity and 30% query coverage indicates that we have found a common domain between the two sequences that spans 30% of their length.

The second parameter is the “E-value” - The Expected value (E) is a statistical parameter that describes the number of hits one can "expect" to see by chance when searching a database of a particular size with a query of a given length. Essentially, the E value describes the random background noise.

- The lower the E-value, or the closer it is to zero, the more "significant" the match.

- Short alignments have relatively high E values. This is because the calculation of the E value takes into account the length of the query sequence. These high E values make sense because shorter sequences have a higher probability of occurring in the database purely by chance.

The last part of the result page presents the alignment of each result with the query sequence. All output parameters are also presented. Please note that the percentage value given at the output table presents only the percentages of identity, but in the description of the actual alignment you can see the percentage of similarity as well.

In general, when filtering our results, we should look for matches with high query coverage and high similarity, alongside a very low e-value. It is also important to look at the alignment itself to see that the sequences are well aligned.

Now let’s go back to the results table. Click on the coverage column header to sort the table by coverage. The first result has a 100% of identity and 100% of coverage, this means that we have found our unknown protein. Click on the first result description in order to find out what this protein is. We

can see in the alignment description that it is the Insulin protein. As the organism name is not mentioned in the description we will click on one of its accession numbers and reveal that it comes from pink salmon.

Another example of using blast is to find pairs of homologous proteins between two organisms. For example, the mouse homolog of the human 1L1B. In order to find a protein homolog from another organism we need to run blast reciprocally. It is not enough to run blastp and find the best protein match in the other organism, as being a homolog requires that the human protein will find a mouse protein as its best match, and that when this mouse protein is used as the query, then the original human protein will be its best match. The reason for this requirement is that many genes have paralogs, related genes that evolved via duplications, and thus finding the exact actual homologous pairs may be tricky. We will run an example to make it clearer. Let's find the mouse protein homolog of the human IL1B. At first, we will run blastp on the human IL1B sequence against the mouse proteome. Open the Blastp page and enter the human IL1B sequence at the sequence window.

```
>NP_000567.1 interleukin-1 beta proprotein [Homo sapiens]
MAEVPELASEMMAYYSGNEDDLFFEADGPKQMKCSFQDLDLCPLDGGIQLRISDHHYSKGFRQAASVVVA
MDKLRKMLVPCPQTFQENDLSTFFPFIFEEEPIFFDTWDNEAYVHDAPVRSLNCTLRDSQQKSLVMSGPY
ELKALHLQGQDMEQQVVFSMSFVQGEESNDKIPVALGLKEKNLYLSCVLKDDKPTLQLESVDPKNYPKKK
MEKRFVFNKIEINNKLEFESAQFPNWYISTSQAENMPVFLGGTKGGQDITDFTMQFVSS
```

Write mus musculus at the organism window in order to run it against the mouse proteome and no other organisms. Click on Blast to run it. At the results page, sort the results by query coverage and choose the best match from the mouse proteins results list. This mouse protein will be defined as a potential homolog to the human protein. Copy its accession number. Now we need to run blastp again, however, this time with the mouse potential homolog protein against the human proteome. Open the blastp page again. Enter the mouse accession number at the sequence window and write homo sapiens at the organism window. Finally, click on blast to run it. At the blast result page, we got the IL1B as the best blastp match and therefore we can define the mouse protein as the mouse homolog of the human IL1B.