keystone

UNIVERSITY OF MALTA
Faculty of Information and
Communication Technology

1st KEYSTONE Training Summer School

Hackathon Session

23rd - 24th July 2015 - University of Malta

Jean-Paul Ebejer,Joel Azzopardi, Joseph Bonello, Charlie Abela

Contact info: jean.p.ebejer@um.edu.mt

This data hackathon consists of processing "large" files consisting of biological data to extract useful information.

Challenge Overview

You will be given a text file containing the complete human genome (`GRCh38.p3.genome.fa`, 3.2GB), and an index file (`gencode.v23.annotation.gff3`[1], 1.2GB) which contains annotations of features in that genome. These files have been downloaded from the GenCode project[2]. This `.fa` file (known as a FASTA file) contains genomic sequence data (bases G, T, A, and C) for each chromosome. A FASTA file is typically made up of a number of records, each consisting of a header followed by a sequence. The chromosomes are therefore defined with a header line which starts with a **'>'** character (e.g. `>chr10 10`). The genomic sequence of the chromosome starts in the next line after the header and continues until either another header line is found or the end of file is reached.

BRCA1 and BRCA2 are two genes that code for tumour suppression and DNA repair proteins. When a mutation occurs in these two genes, the production of these proteins may be affected negatively. This increases the likelihood of developing breast and ovarian cancer in females. In this hackathon you will deal with data regarding these genes.

[1] The GFF3 file format is described here: http://www.gencodegenes.org/data_format.html

[2] http://www.gencodegenes.org/

Tasks

The programming tasks will focus around the gene BRCA2 (gene identifier: ENSG00000139618.14) which is located on chromosome 13.

1. Using the index file as a guide, find the protein-coding sequences for the BRCA2 gene and extract these (displaying them to the user). The chromosome number (column 1) in the annotation file (.gff3) should be "chr13" for chromosome 13. The feature type (column 3) in the annotation file should be 'CDS' for coding sequence. The ninth column in the annotation file contains key-value pairs with additional information. Using this field extract the entries for BRCA2 (where gene_id is "ENSG00000139618.14" and transcript_type is "protein_coding"). Use the start (column 4) and end (column 5) values in the annotation file to extract the BRCA2 DNA sequence from the FASTA file. In order to do this, you need to first extract the sequence of chromosome 13 from the FASTA file. The header line for this entry is ">chr13 13". (10 points)
2. Translate the genomic sequence you have extracted above into the corresponding protein sequence. This occurs by reading in triplets of DNA bases (known as codons) and translating these into amino acids[3]. The genomic phase (Column 8) in the annotation file determines the starting offset to read the sequence in. You can test the generated (translated) protein sequence by using it as a query in BLAST[4] using protein blast (blastp) and "Homo Sapiens" as the target organism. The BRCA2 protein should be your top result hit. (10 points)
3. In previous tasks we have searched for the sequence given the gene identifier (or name) and the chromosome where the gene is located. Your task is now to execute a reverse search - given a genomic sequence, find the possible gene name(s) and chromosome location. In this task we assume exact matches between the query and the sequence. The result should also show the total number of matches of the query to the genome. (10 points)
4. Generate a simple visualization which shows where these genomic queries are foundin the human genome. (10 points)

   This could look something like the following (where each dash character represents 10,000,000 genomic bases):

```
         *    * *          **
chr1  -------------------------
       *              *
chr2  -------------------------
...
         *
chr14 ----------
...
```

[3] The DNA-to-Amino Acid translation table may be found here: https://en.wikipedia.org/wiki/DNA_codon_table

[4] http://blast.ncbi.nlm.nih.gov/

5. As an added task, allow for fuzzy searches. A five-letter genomic query 'GATAG' with 1 mismatch allowed will also match 'AATAG' and 'GATAA' but not 'AATAA'. The number of mismatches allowed should be parameterized and cannot be more than 70% of the length of the query. (10 points)
6. You have been given a protein sequencein file protein.txt. Find the genomic counterpart in the genome FASTA file. (10 points)

The boring bit: Judging Criteria, Rules and Prizes!

The idea behind this competition is for students to apply the theoretical concepts they have learnt during the summer school to a practical scenario. The following points apply to the competition:

- Participation to this hackathon is reserved to students attending the 1st Keystone Training Summer School at the University of Malta.
- Participation is limited to groups of at least two and at most three students. Registrations should be made by emailing: jean.p.ebejer@um.edu.mt
- Prizes are of the order of €100 for each participant in the group that places first; €70 for those placing second; and €40 for those placing third.
- At the end of the Hackathon each group will have to present their approaches (design decisions etc.) in a five minute presentation.
- Also, a live demonstration of the working tasks has to be given to one of the judges.
- Points will be allocated not only for the implementation of the specified tasks, but also based on execution speed of the solution and on the most innovative ideas when tackling "big" data.
- For this hackathon you may use any OS and programming language of your choice.
- Each group will be given a Microsoft Azure account. You can start as many virtual machines as you like, but the instance type is required to be A3[5].
- Finally, the judging panel's decision is final.

Sponsors

ALTARO

Microsoft

cost
EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

COST is supported by the EU
Framework Programme
Horizon 2020

[5] http://azure.microsoft.com/en-us/pricing/details/virtual-machines/