Exercise 2 : Identifying Repeats in genomic sequence with RepeatMasker

General note : All programs used in this workshop are installed in /usr/local/ensembl/bin, which is already added to your $PATH variable. We decided to use tcsh as default shell - if you'd like to use a different shell please feel free to do so and we'll try to support you wherever we can.

RepeatMasker is one of the algorithms that we use to mask out repetitive sequence during the genebuild process. In this exercise we will run RepeatMasker on some genomic mouse sequences manually.

- You will find example sequences in these folders :

/home/training/ensembl-data/genebuild/assembly/
/home/training/ensembl-data/genebuild/assembly/test_seqs

- You will find more information and an FAQ about RepeatMasker at http://www.repeatmasker.org/
- Running RepeatMasker without arguments will give you a list of options to use
- An example commandline can look like this :

```
perl /usr/local/ensembl/bin/RepeatMasker_3_1_8/RepeatMasker \
-species <speciesname> -qq \
-dir <full_path_to_output_directory> <full_path_to_input_sequence>
```

- Before running RepeatMasker, create an output directory for the results:

```
mkdir -p /home/training/ensembl-data/genebuild/output/repeatmask_output
```

- Depending on the length of your input sequence, the run can take a few minutes. You can identify the length of a fasta sequence with the program fastalength.

```
fastalength \
/lustre/work1/ensembl/jhv/project_workshop/cshl_workshop/2_seq_data_loa
ding/test_seqs/test_sequence_to_repeatmask.fa
```

- Now RepeatMask the test sequence (will take about a minute) by pasting the following command in your terminal:

```
perl /usr/local/ensembl/bin/RepeatMasker_3_1_8/RepeatMasker \
-species mouse -qq \
-dir /home/training/ensembl-data/genebuild/output/repeatmask_output \
/home/training/ensembl-
data/genebuild/assembly/test_seqs/test_sequence_to_repeatmask.fa
```

Take a look at the output files produced by the RepeatMasking run and check how much of the sequence has been masked, plus which repeat types have been identified - you will find those files in your output-directory

```
(/home/training/ensembl-data/genebuild/output/repeatmask_output).
```

```
ls  /home/training/ensembl-data/genebuild/output/repeatmask_output
```

```
chromosome:NCBIM37:11:69703858:69950060:1.fa.cat :
```

(different format of outfile)

```
chromosome:NCBIM37:11:69703858:69950060:1.fa.log
```

(a log)

```
chromosome:NCBIM37:11:69703858:69950060:1.fa.masked
```

(hardmasked fasta file where REPEAT is replaced by NNNNNNNN)

```
chromosome:NCBIM37:11:69703858:69950060:1.fa.out
```

(list of identified repeats and their positions)

```
chromosome:NCBIM37:11:69703858:69950060:1.fa.tbl
```

(summary of identified repeat-types, percentage.)

* * * End of Exercise * * *

Notes on using MySQL

Ensembl uses open source software wherever possible and this is one of the main reasons that we have chosen MySQL. Although we usually access the database through the Ensembl API, we sometimes also need to query our databases by hand. For this it is useful to have a good understanding of the Ensembl database schema, particularly for the tables that are useful to you.

To connect to a database, type:

```
mysql -uens-training -pworkshop -htrpc6c04 -P3306 -D usrname_mouse37_ref
```

You can now type in your commands. Each line should end with a semi-colon ';'.

To connect to a different database, type:

```
use usrname_mouse37_other_database;
use usrname_mouse37_ref;
```

To see all tables in the database, type:

```
show tables;
```

To look at the schema for a particular table eg. analysis table, type:

```
describe analysis;
```

To look at all data in the analysis table, type:

```
select * from analysis;
```

To look at only 5 rows in the analysis table, type:

```
select * from analysis limit 5;
```

To look at the analysis_id and logic_name columns of the analysis table (all rows), type:

```
select analysis_id, logic_name from analysis;
```

When you're finished and would like to disconnect from your database, type:

```
quit; (or exit;)
```

Most commands can also be executed as follows:

```
mysql -uens-training -pworkshop -htrpc6c04 -P3306 -D usrname_mouse37_ref -e "describe analysis;"
mysql -uens-training -pworkshop -htrpc6c04 -P3306 -D usrname_mouse37_ref -e "select * from analysis;"
mysql -uens-training -pworkshop -htrpc6c04 -P3306 -D usrname_mouse37_ref -e "select analysis_id, logic_name from analysis;"
```