

```

#! /usr/local/bin/perl -w

use strict;
use warnings;

use lib '/project/rodent/medaka_anja/pscripte';
use ClusterFunc;

my($USAGE) = "$0  ***Plattenanzahl oder Sequenanzahl/400***\n";

unless (@ARGV) {
    print $USAGE;
    exit;}

my $num = $ARGV[0];
my $plate=0;
my $output='';
my $files='';
my $vector = 'pCS2_vector_fasta';
my $crossmatch_info = '';

#Sequenz aus abi files mit extract_seq extrahieren + Crossmatch
#for ($plate=1;$plate<($num+1);++$plate)
#  {$output='plate'.$plate.'.fasta';
#   if ($plate < 10) {$files='McF000'.$plate.'*.ab1';}
#   else {$files='McF00'.$plate.'*.ab1';}
#   system "extract_seq -fasta_out -output $output $files";
#   $crossmatch_info = 'screen'.$plate.'.out';
#   system "touch $crossmatch_info";
#   system "cross_match $output $vector -minmatch 10 -minscore 20 -
screen >$crossmatch_info";
#}
#my $crossmatch_output = 'plate1-'. $num.'.screen';
#system "cat plate*.fasta.screen >>$crossmatch_output ";

my $crossmatch_output = 'seqs_fasta.screen';

#FASTA file in DB umwandeln
my $BlastDB = $crossmatch_output."_db";
#system "formatdb -p F -i $crossmatch_output -n $BlastDB >!
formatDB.out";

#Blast der Sequenzen gegeneinander
my $BlastResult = $crossmatch_output."_blasteach";
#system "blastall -p blastn -i $crossmatch_output -e 1.0e-50 -d
$BlastDB -o $BlastResult";

#Edit_blasteach; Editieren des Blastoutputs; *.edit wird geschrieben
#Edit_blasteach($BlastResult);

#*.screen in einzelne fasta files splitten, damit später Gruppen neu
#zusammengefügt werden können
#Split_fasta($crossmatch_output,0);
#my $qual_output = 'seqs_fasta.screen.qual';
#Split_fasta($qual_output,1);

#Group_fasta fasst fasta files zu *.group zusammen; gleichzeitig wird
fofn_group geschrieben
my $Group_input = $BlastResult.'.edit';
my $Group_output = 'fofn_group';
my $Qual_output = 'fofn_qual';
#Group_fasta($Group_input);

#Run_CAP3($Group_output);

```

```
#Edit_group($Group_output);  
#Edit_CAP3();  
Edit_OutputCAP3('CAP3_out_seqsfasta',0);  
exit;
```