

ESIL 2^e année. Examen Unix/Perl

Avril 2002

Durée: 2 heures

Daniel GAUTHERET

Documents autorisés

1. (6 pt)

Identifiez parmi les commandes Perl suivantes celles qui comportent une erreur. Corrigez l'erreur.

- a. `print "Can't open \"$ARGV[0]\"\\n";`
- b. `while (($line = <F>) !~ /^>/) {}`
- c. `if (@line[$i]=~ /Query\:\s(\d+).* (\d+)$/) {`
- d. `@AUs = ($seg =~ /[CAN]/g);`
- e. `if (($totseq % 100)= 0){`
- f. `$ajoutA = $doublet .= "A" ;`

3. (8 pt)

Considérez le programme Perl en Annexe 1, nécessitant des fichiers au format EMBL comme celui de l'annexe 2.

- a. A quoi sert ce programme ?
- b. Quel format de séquence est adopté en sortie du programme ?
- c. Quelle est la fonction de la partie de code marquée « 1 » ?
- d. Quelle est la fonction de la partie de code marquée « 2 » ?
- e. A quoi sert la variable \$cutseq ?
- f. Expliquez à quoi sert la ligne :
`} elsif ($line =~ /^\\\/\\\/) {`
- g. Quel serait la sortie du programme si on lui fournissait en argument le nom du fichier en annexe 2 ?

4. (6 pt)

On dispose d'un tableau associatif **%scores**, donnant le score de tous les dinucleotides possibles (« AA », « AT » etc.) dans un modèle de Markov d'ordre 1. Ecrivez un programme qui permet de rechercher dans une séquence au format Fasta la sous-séquence de longueur L ayant le score le plus élevé selon ce modèle.

ANNEXE 1

```
#!/usr/bin/perl

if (!open(F, $ARGV[0])) {
    print "Can't open \"$ARGV[0]\"\n";
    exit;
}

while ($line = <F>) {
    chop $line;
    if ($line =~ /^ID +(\w+)/) {
        $id .= $1;
    } elsif ($line =~ /^DE +(.)+/) {
        $de .= $1;
    } elsif ($line =~ /^FT +CDS +(\d+)/) {
        $cd = 1;
        $cdspos=$1;
    } elsif ($line =~ /^\\\/\\\/) {
        if ($cd) {
            $start=$cdspos-100;
            if ($start > 0) {
                $cutseq=substr($seq,$start,100);
                print ">$id $de\n" ;
                for( $i = 0 ; $i < length($cutseq) ; $i+=80){
                    print substr ($cutseq, $i, 80), "\n";
                }
            }
        }
        $id="";
        $de="";
        $cd=0;
        $cdspos=-1;
        $seq="";
    } elsif ($line =~ /^
        /) {
        $line =~ s/[ 0123456789]//g;
        $seq .= $line;
    }
}

```

ANNEXE 2

```
ID AF043732 standard; RNA; HUM; 894 BP.
AC AF043732;
SV AF043732.1
DT 17-AUG-1998 (Rel. 56, Created)
DT 03-MAR-2000 (Rel. 62, Last updated, Version 4)
DE Homo sapiens cGMP-binding cGMP-specific 3',5'-cyclic nucleotide
DE phosphodiesterase (PDE5A2) mRNA, partial cds.
OS Homo sapiens (human)
OC Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia;
OC Eutheria; Primates; Catarrhini; Hominidae; Homo.
RP 1-894
RT cGMP-binding, cGMP-specific 3',5'-cyclic nucleotide phosphodiesterase";
RL Gene 216(1):139-147(1998).
FH Key Location/Qualifiers
FH
FT source 1..894
FT /chromosome="4"
FT /db_xref="taxon:9606"
FT /organism="Homo sapiens"
FT CDS 469..>894
FT /codon_start=1
FT /gene="PDE5A2"
FT /product="cGMP-binding cGMP-specific 3',5'-cyclic
FT nucleotide phosphodiesterase"
FT /translation="MLPFGDKTREMVNWAFAERVHTIPVCKEGIRGHTESCSCPLQQSP
FT RADNSVPGTPTRKISASEFDRPLRPIVVKDSEGTVSFLSDSEKKEQMLTPPRFDHDEG
FT DQCSRLLLELVKDISSHLVDVTALCHKIFLHIHGLISADR"
SQ Sequence 894 BP; 175 A; 253 C; 284 G; 182 T; 0 other:
gaattccggg tggagcagag agtgcttggc gggcgggtgc cccgaagcct gaggaattga 60
tgcatagtcg ggggtgctgg actccaggac ggcgcccttt gcagaagag tccgtgcccc 120
gctgccgctt gctctgggca gggccgctcg ggcagcgggt gctggccgtc ttggccggga 180
gggtgacaca cgcccgccag tgatggtcga ggagcccgcc cgggttcgac gcccggttcc 240
cactcgccgg cgcgcggccg cggctctgcc ggtcgctccc cgccaggggg gcggcgcaagc 300
gcggggcgag cgggagcgca cgggcggcgg cgcggaggcc ggggcccgtga gggagctggt 360
tctgagtcgc cgacctcacc tcacctcacc tgcgcgctgt gttgccccta cggagctccc 420
tggagtcag gatcgccgga gttcgaaaac gaacttccc cgtttgctat gttgcccttt 480
ggagacaaaa caagagaaat ggtcaatgca tggtttgctg agagagttca caccatccct 540
gtgtgcaagg aaggtatcag aggccacacc gaatcttgct cttgtccctt gcagcagagt 600
cctgtgtagc ataacagtg ccttggaaac ccaaccagga aaatctctgc ctttgaattt 660
gaccggcctc ttagacctat tgttgtcaag gattctgagg gaactgtgag cttctctctc 720
gactcagaaa agaagaaaca gatgcctcta accctccaa ggttgatca tgatgaaggg 780
gccttatgtc acaaaatctt cttgcatatc catggactga tatctgctga ccgc 894
//

```