BDF Hackathon

Vision

The aim of this hackathon is to come up with algorithms to solve problems in biology and biocomputation. The organisers feel that programming is in everyone's blood, for we are inherently algorithmic. Hence, the questions do **not** expect that you know some fancy set of functions to solve these problems. They just need you to think about how you would go about doing something.

Most of the questions are **not** inherently biological in nature. Biology however does provide motivation to solve these problems. The organisers feel that these questions do not need any particular knowledge of Biology (partly because the organisers themselves are not great fans of Biology). Theory and necessary references are linked with the questions.

Feel free to call the volunteers if you need additional information.

Click here for Rules

1. DNA Base counting

Main question

Shravan Sharath asks you to find the number of As, Ts, Gs, and Cs in the DNA genomic sequence of the coding strand in your BIO 101 Class,here

Side question

Verify if Chargaff's Rule holds true for this sequence. If it is not, why not? Assume that this is in fact a real DNA.

Example

Input

CATGGTCATTTATCCTCTAATATATGGCCCGCACTTAGGCAAAGGTTGCACTCTAGCGCATGCGTGTTTAAGTTTAGCCGTTAGGCGTGCGGGGATCCGA

Output

A=21,T=29,G=27,C=23

2. Protein making

Main question

Samarth Mukhopadhyay has given you a DNA sequence of a template strand, here, first transcribe it to the corresponding RNA sequence and then use the following table to translate it to the Primary Protein sequence. Assume that 3 RNA bases code for one Amino Acid, and that translation starts from the first base itself. Do not assume stop codons.

Table -

{"AAA"=>"A01", "UAA"=>"A02", "GAA"=>"A03", "CAA"=>"A04", "AUA"=>"A05", "UUA"=>"A06", "GUA"=>"A07", "CUA"=>"A08", "AGA"=>"A09", "UGA"=>"A10", "GGA"=>"A11", "CGA"=>"A12", "ACA"=>"A13", "UCA"=>"A14", "GCA"=>"A15", "CCA"=>"A16", "AAU"=>"A17", "UAU"=>"A18", "GAU"=>"A19", "CAU"=>"A20", "AUU"=>"A21", "UUU"=>"A22", "GUU"=>"A23", "CUU"=>"A24", "AGU"=>"A25", "UGU"=>"A26", "GGU"=>"A27", "CGU"=>"A28", "ACU"=>"A29", "UCU"=>"A30", "GCU"=>"A31", "CCU"=>"A32", "AAG"=>"A33", "UAG"=>"A34", "GAG"=>"A35", "CAG"=>"A36", "AUG"=>"A37", "UUG"=>"A38", "GUG"=>"A39", "CUG"=>"A40", "AGG"=>"A41", "UGG"=>"A42", "GGG"=>"A43", "CGG"=>"A44", "ACG"=>"A45", "UCG"=>"A46", "GCG"=>"A47", "CCG"=>"A48", "AAC"=>"A49", "UAC"=>"A50", "GAC"=>"A51", "CAC"=>"A52", "AUC"=>"A53", "UUC"=>"A54", "GUC"=>"A55", "CUC"=>"A56", "AGC"=>"A57", "UGC"=>"A58", "GGC"=>"A59", "CGC"=>"A60", "ACC"=>"A61", "UCC"=>"A62", "GCC"=>"A63", "CCC"=>"A64"}. //You may directly copy paste this as necessary.

Side question

The total number of possible Amino acids are 64, as we saw above. However only 21 are actually found in natural settings. What could be the biochemical/evolutionary reason for this?

Example

Input

TGTCGACATGTTGCCCCCTCAACTCTCTGT

Output

A16A37A59A45A41A57A07A39A26A26

3. Evolutionary simulation

Main question

Nagesh Giri Pramesh wants you to simulate Selection in evolution in the following way.

- Consider a 256x256 matrix of "Ground". The ground color gradually transforms from black to white
- Each cell hosts a *diploid* individual of a species which has a very peculiar genetic inheritance-
  - It has 256 body color loci.
  - Each loci has only 2 forms, colored and uncolored with the *colored being dominant.*
  - The phenotype of the individual is the number of colored loci.
- Initialize the environment
- A generation cycle is a cycle of death and reproduction.
  - Death happens only by an aerial predator.
    - ⋆ The difference between the color of the ground and individual increases the chances of being eaten *linearly.*
    - ⋆ The maximum chance of death is 90% and minimum 5%.
    - ⋆ The cell which the individual inhabited becomes "empty" when the individual dies.
  - Reproduction can happen in one of 2 ways.
    - ⋆ For every cell, an adjacent cell is chosen.
    - ⋆ If the adjacent cell is empty, the individual is "copied"
    - ⋆ If the adjacent cell is not empty, the two individuals mate, and 2 progeny are created. These progeny have loci values randomly chosen from the parents (one each).

            - Eg - For *1* locus, if one parent was Cc and the other cC, the progeny *may* get Cc or cc or CC in the expected
        - Assume law of independent assortment
        - Assume law of segregation
        - These two progeny are then placed in place of the parents
- Make an image of the individuals on the ground after each generation cycle and save the image. Do this for 100 generations. Make a gif of these from an online gif maker, or from within your language of choice itself.
    - Note - Animations in Python are pretty messy.

Hint

- The number 256 is chosen for a reason.

Side question

What assumptions have we made in the above simulation? How valid are these assumptions in a real-life scenario?

4. Viral relatedness

Ritajyoti Ray Choudhary has given you two RNA sequences of different viruses here, find how long ago in hours they diverged from their common ancestor. Assume the following -

- All variation is due to genetic mutations only.
- Mutations are only of the point-type mutations.
    - Insertion of a single base.
    - Deletion of a single base.
    - Substitutions of a base. Transversions and transitions are equally probable.
- Mutations occur at a rate of 0.1% of total genome per generation.
- All mutations are neutral.
- Assume an insertion and Deletion in one location as a single Substitutions.
- Assume double Substitutions as single Substitutions.
- 1 generation lasts 3 hours.

Side question

What assumptions have we made in the above simulation? How valid are these assumptions in a real-life scenario?

Example

Input

GACAUGUCUC, GCAUCGUAUC

Output

900

Because - (This is not required to be outputted)

Number of mutations - 3

- delete A at pos 2
  - G ~~A~~ CAUGUCUC
- insert C at pos 5
  - GCAU **C** GUCUC
- Substitute C->A at pos 8
  - GCAUCGU ~~C~~ **A** UC

Mutations per generation = 0.01 generations for one mutation = 100

Length of a generation = 3 hours

Hence time of divergence = 3 mut * 100 gen/mut * 3 hr/gen = 900 hr

5. Cumulative Species Accumulation curve.

Ranjani Jain has given you birding data of an individual as a csv from her Minivet Birding app, plot the Cumulative Species Richness/Accumulation Curve. In case you were sleeping in her class, look here.

Example Data

Name	Genus	Species	Session Name	Session Is Complete	Session Number	Session Remarks	Type	Latitude	Session Start Time	Longitude	Accuracy	Zone	Date	Time
nia	Prinia	socialis	session1	Yes	1		TRAVELING	30.66[illegible]	1/17/2[illegible]	76.729876	-1	CAF	1/17/2020	19:27
nia	Prinia	socialis	session1	Yes	1		TRAVELING	30.66[illegible]	1/17/2[illegible]	76.729876	-1	CAF	1/18/2020	16:27
nia	Prinia	socialis	session1	Yes	1		TRAVELING	30.66[illegible]	1/17/2[illegible]	76.729876	-1	CAF	1/19/2020	19:32
nia	Prinia	socialis	session1	Yes	1		TRAVELING	30.66[illegible]	1/17/2[illegible]	76.729876	-1	CAF	1/19/2020	20:45

6. Bird call counting

Assume that every bird song is made up of these following constituent sub-calls-

- Chirp (chp)
- Whistle (wsl)
- Trill (trl)
- Click (clk)
- Buzz (bzz)
- Rest (rst)

An analysis of world bird data tells us the following

- If difference in call from a standard call are less than 5% apart, they are always of the same species.
- If difference in call from a standard call are more than 10% apart, they are always of different species.
- If difference is between 5-10%, one cannot be sure.

Bird and Environment Lab (BEL), IISER Mohali has recorded sound from a forest for 15 minutes and individual bird calls were isolated, which can be found here.

The following is a list of the calls of all birds that were made by BEL members from actual observation **manually**.

(You may copy-paste this)-

- Fulvous whistling duck : wsl-bzz-chp-trl-wsl-clk-trl-trl-chp-bzz-wsl-bzz-wsl-wsl-wsl-wsl-trl-wsl-trl-chp
- Lesser whistling duck : wsl-rst-trl-clk-bzz-chp-wsl-rst-trl-trl-clk-wsl-clk-chp-rst-clk-clk-bzz-bzz-clk
- Red-breasted goose : trl-wsl-wsl-wsl-rst-chp-trl-wsl-clk-rst-clk-wsl-chp-clk-bzz-clk-bzz-wsl-wsl-trl
- Bar-headed goose : wsl-wsl-trl-trl-clk-trl-clk-clk-rst-wsl-bzz-rst-trl-bzz-clk-chp-wsl-bzz-bzz-clk
- Graylag goose : chp-bzz-bzz-bzz-clk-wsl-bzz-trl-bzz-chp-bzz-wsl-clk-wsl-bzz-trl-clk-rst-trl-bzz
- Taiga bean-goose : clk-bzz-wsl-rst-wsl-bzz-chp-rst-wsl-rst-wsl-clk-trl-bzz-chp-trl-clk-rst-wsl-wsl
- Tundra bean-goose : trl-clk-bzz-rst-wsl-bzz-wsl-trl-bzz-wsl-rst-rst-rst-rst-wsl-chp-rst-clk-clk-trl
- Greater white-fronted goose : rst-chp-chp-wsl-chp-wsl-chp-rst-rst-wsl-bzz-clk-clk-wsl-clk-wsl-clk-trl-wsl-clk

- Lesser white-fronted goose : clk-chp-bzz-chp-clk-bzz-chp-chp-wsl-wsl-clk-rst-rst-wsl-wsl-clk-rst-wsl-trl-bzz
- Mute swan : wsl-trl-trl-bzz-trl-trl-bzz-bzz-rst-clk-trl-clk-chp-trl-clk-wsl-chp-trl-bzz-trl
- Tundra swan : clk-wsl-bzz-trl-clk-rst-rst-rst-clk-trl-wsl-rst-wsl-chp-clk-clk-clk-clk-chp-chp
- Whooper swan : rst-rst-clk-clk-bzz-rst-wsl-bzz-trl-bzz-bzz-rst-chp-trl-wsl-trl-bzz-bzz-wsl-chp
- Knob-billed duck : rst-clk-chp-trl-clk-wsl-chp-rst-clk-bzz-chp-trl-bzz-bzz-chp-trl-chp-rst-bzz-trl
- Common shelduck : trl-trl-trl-trl-wsl-chp-trl-trl-trl-trl-trl-chp-bzz-wsl-clk-rst-trl-trl-rst-rst
- Ruddy shelduck : clk-trl-rst-chp-trl-bzz-bzz-clk-clk-bzz-rst-bzz-rst-chp-bzz-bzz-rst-trl-trl-trl
- White-winged duck : clk-chp-bzz-chp-chp-chp-chp-bzz-clk-trl-trl-rst-trl-trl-wsl-chp-clk-trl-wsl-chp
- Mandarin duck : bzz-bzz-chp-wsl-rst-chp-wsl-clk-rst-clk-chp-rst-wsl-bzz-trl-rst-bzz-wsl-rst-chp

Count the following -

- Number of species in data
- Number of individuals of each species
- Number of identifiable species. First verify if BEL data is correct. Members of BEL are known to be lax in their work. If you feel that two birds may be the same species, assume the species that occurs first in the above list.

Side Question

How valid is this sort of analysis to ID birds on the basis of the call. What would be the problems one may face when trying to do this? You may want to look at PCA.