

Gene Processor

Have you ever wondered why some have blue eyes while others have brown? Or perhaps why some are shorter than others?

You may have heard that genes dictate one's traits. Specifically, genes make proteins that give shape to our personality and appearance.

In this program, the user will input a gene. The program will convert it to the necessary molecules. And finally, it will output the corresponding proteins.

What are genes?

Genes are sections of your DNA. Each part of the gene consists of one of the following molecules: adenine, cytosine, guanine, thymine and uracil.

These molecules can be represented as the characters A, C, G, T and U respectively. So for example, a gene can have the following molecules: TAG GCT TCA CAG GCC.

How does one produce proteins?

Genes from your DNA are arranged according to trios known as triplets. To make proteins, these triplets must be converted to codons. Once the codons are converted, these molecules will be sent outside the nucleus to form the proteins.

Protein-making at this point is a lot like matchmaking! The codons attach to another set of trios known as anticodons. These trios contain the building blocks of protein known as amino acids.

Converting from triplets to codons to anticodons

Converting the triplets is simple! Just match their letters to their corresponding "partner". The following shows the conversion of each molecules:

| From Triplet | To Codon | To Anticodon |
|--------------|----------|--------------|
| T | A | U |
| A | U | A |

| | | |
|---|---|---|
| G | C | G |
| C | G | C |

For example, the user inputs the triplets: ACT CGG GAT. The corresponding codon is UGA GCC CUA. The corresponding anticodon is ACU CGG GAU.

Remember that codons and anticodons don't have the molecule thymine. Instead, they have uracil. Conversely, a normal triplet doesn't have any uracil.

Making the actual protein

For the sake of simplicity, assume there are only five proteins. Simply match the anticodon to its corresponding protein. If no protein can be found, the gene is said to be non-functional.

| From Anticodon | To Protein | Display As |
|----------------------|----------------|------------|
| AUG | Methionine | Met |
| GCC | Alanine | Ala |
| UGC | Cystine | Cys |
| AAA | Lysine | Lys |
| ARG | Arginine | Arg |
| If not seen above... | Non-functional | --- |

To-Do

We defined **an array of three (plus one null byte) character** as a Triplet. A Gene is defined as **an array of Triplets**.

The variable AminoAcid is **an array of three (plus one null) characters** while a Protein type is an array of Amino Acids.

Your goal now is to convert the normal triplets to codons then to anticodons. From there, match the anticodons to their corresponding amino acids.

In genome.c

☐ Finish the `print_genome()` function.

Should accept a Gene and max number of triplets as its parameter. The function should print the genome in the following format: xxx xxx xxx xxx xxx.

For example, AUG GCC UGC AAA AUG.

☐ Finish the `print_protein()` function.

Should accept a protein and max number of amino acid as its parameter. The function should print the genome in the following format: xxx - xxx - xxx - xxx - xxx.

For example, MET - ALA - --- - LYS - CYS (the third amino acid is non-functional).

☐ Finish the `get_codon()` function.

Should accept two Triplets as a parameter. The second triplet is the normal triplet that will be converted to its codon counterpart. Its codon counterpart will be stored in the first Triplet.

For example:

Let the variable `triplet` be "ACT" and `codon` be its corresponding counterpart. Then, `get_codon(codon, triplet)` will modify the `codon` variable to be equal to "UGC".

☐ Finish the `get_anticodon()` function.

Should accept two Triplets as a parameter. The second triplet is the codon that will be converted to its anticodon counterpart. Its anticodon counterpart will be stored in the first Triplet.

For example:

Let the variable codon be "UGC" and anticodon be its corresponding counterpart.
Then, `get_anticodon(anticodon, codon)` will modify the anticodon variable to be equal to "ACG".

- ☐ **Finish the `get_amino_acid()` function.**

In `genomeMain.c`

- ☐ Complete the arguments in `scanf` for scanning the genes.
- ☐ Call the `print_genome()` function to print .
- ☐ Basta tignan niyo lang sa program kaloka.