# IDbyDNA Candidate Challenge – Software Engineer

Please complete and answer the following questions.

Keep in mind that we don’t want to see unnecessary complexity in the answers – please answer as clearly and concisely as possible. The code itself should demonstrate what you consider to be production level quality.

When submitting your code challenge solution, include complete command line instructions for building, and launching the application. Also consider that additional tests may be run against the code that you provide.

## Question 1

Download a sample dataset using this link:

<https://drive.google.com/file/d/1zYt0XHzPOI37klCEpy0pjQIBpaf5wrz4/view?usp=sharing>.

This is a FASTA file (for explanation of the format see <https://en.wikipedia.org/wiki/FASTA_format>). The first entry is:

>SRR1748776.1 1 length=251

CGGTTCAGCAGGAATGCCGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGATAGCGATCTCGT

ATGCCGTCTTCTGCTTGAAAAAAAAAGACAAGGCTCCTGAATTCGCGTCTGCATATCGGGTGACCATCCC

CCAAGGCCTAATCCGCCAACCTGACCGACAGCGATCCATTACCGCGAGGGAAAGGCGCTACTACCCCCTG

TGAGGTCAGCGAACCAGATCCTTACACCGGATCGGTATAGC

The line that starts with a ‘>’ is called the header. The lines that follow are the sequences. Even though there can be line breaks in the sequence, it is actually to be interpreted as one long string. In this sample we call each of the header/sequence blocks a *read*. There are 130016 reads in the file you have downloaded. All the reads in this sample are the same length of 251 base pairs (letters).

A k-mer is a substring of length *k*. For example, the first 6-mer of CGGTTCAGCAGGAATGCCGAGATCGGAAGA would be CGGTTC. The second 6-mer would be GGTTCA, the third 6-mer would be GTTCAG, etc.

Implement your own hash table to count all of the k-mers of length 25 (25-mers) in the downloaded reads. Exclude any k-mer that contains a letter that is not A, T, C, or G. When you implement your own hash table, you cannot use any third-party hash tables, or hash tables that are part of the language, or standard library, but you can use a third-party hashing function.

1. How many distinct k-mers are there?
2. What is the total number of k-mers?
3. Which k-mer has the highest count?

Provide your hash table code as an attachment.

## Question 2

As part of a larger application that needs access to some input data, create a good design-centered solution that will accept input in multiple data formats, and convert them to a common format. The format of the input data will be different, and the data you need to work with always be somewhere inside the data you receive.

The data you are concerned with is the pairing of a string, and its associated number. For example, a pairing might be "A" and 1.

The input you have available to you will be in the following formats. Keep in mind that in the future, there could be 20 completely different input formats that the application would need to handle.

FormatA – text file

A1B2C3D444E55F9

FormatB – JSON file

{

Count: 2,

Parts: ["A", "B", "X", "X"],

Numbers: [1, 2]

}

FormatB – JSON file

{

Count: 3,

Parts: ["A", "B", "X", "X"],

Numbers: [1, 2, 88]

}

The common format of the data the application is expecting is a collection of objects:

CommonFormat:

{Part: "A", Number: 1, PartLength: 1},

{Part: "B", Number: 2, PartLength: 1},

{Part: "CC", Number: 3, PartLength: 2}

How would you structure a solution that would handle these formats, as well as have the ability to handle even more input formats in the future, in a seamless way?

Write an application that would read text files in formatA and formatB and converts the data to the CommonFormat structure. Print out an analysis report that is similar to the following:

A:1 B:2 C:3 D:444 E:55 F:9

A:1 B:2

A:1 B:2 X:88, error found an X

Implementation Notes:

* If there is a pair where the key "X" is mapped to a number, then this is considered an error condition, and there should be some indication that this error condition was detected.
* It is also safe to assume that all strings will be at least one character long.
* You only need to consider valid pairs. E.g., a key with no associated value can be ignored.