# COMP 201 - Spring 2023
# Assignment 1 - Strings in C

**Assigned: 15 March 2023 23:59, Due: 29 March 2023 23:59**

Osman Batur İnce (`oince22@ku.edu.tr`) is the lead person for this assignment.

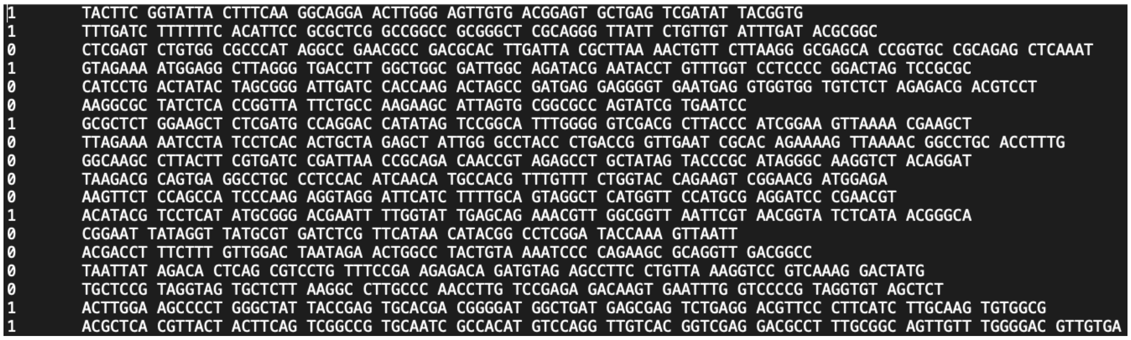
## 1 Introduction

This assignment aims to advance your skills in manipulating strings and learn to use string functions efficiently on C. You will do this by working on a real world problem: identifying cancerous gene sequences. For this, you will use a slightly simplified and different version of **tf-idf** (term frequency - inverse document frequency) score, a widely used natural language processing measure.

You are provided a synthetic dataset, where each line consists of a label indicating whether the sequence on the same line is a cancerous or healthy gene. You will process the `sequences.txt` file and do cancerous gene detection based on calculated **tf-idf** metrics.

## 2 Datasets

`sequences.txt` has classes and gene sequences on each line. A line begins with 0 or 1, where 0 corresponds to the healthy gene and 1 corresponds to the cancerous gene. After the classes, there is a tab (\t) character that separates the classes and gene sequence. After the tab character, the gene sequence starts. The gene sequence consists of multiple words (sequence length $\in \{9, 10, .., 14\}$), where each word has multiple nucleotides (word length $\in \{5, 6, 7\}$). At the end of each line, there is a new line (\n) character. Moreover, we will call **words** for **nucleotide sequences** (i.e. **TACTTC**) for brevity.



```
1	TACTTC GGTATTA CTTTCAA GGCAGGA ACTTGGG AGTTGTG ACGGAGT GCTGAG TCGATAT TACGGTG
1	TTTGATC TTTTTTC ACATTCC GCGCTCG GCCGGCC GCGGGCT CGCAGGG TTATT CTGTTGT ATTTGAT ACGCGGC
0	CTCGAGT CTGTGG CGCCCAT AGGCC GAACGCC GACGCAC TTGATTA CGCTTAA AACTGTT CTTAAGG GCGAGCA CCGGTGC CGCAGAG CTCAAAT
1	GTAGAAA ATGGAGG CTTAGGG TGACCTT GGCTGGC GATTGGC AGATACG AATACCT GTTTGGT CCTCCCC GGACTAG TCCGCGC
0	CATCCTG ACTATAC TAGCGGG ATTGATC CACCAAG ACTAGCC GATGAG GAGGGGT GAATGAG GTGGTGG TGTCTCT AGAGACG ACGTCCT
0	AAGGCGC TATCTCA CCGGTTA TTCTGCC AAGAAGC ATTAGTG CGGCGCC AGTATCG TGAATCC
1	GCGCTCT GGAAGCT CTCGATG CCAGGAC CATATAG TCCGGCA TTTGGGG GTCGACG CTTACCC ATCGGAA GTTAAAA CGAAGCT
0	TTAGAAA AATCCTA TCCTCAC ACTGCTA GAGCT ATTGG GCCTACC CTGACCG GTTGAAT CGCAC AGAAAAG TTAAAAC GGCCTGC ACCTTTG
0	GGCAAGC CTTACTT CGTGATC CGATTAA CCGCAGA CAACCGT AGAGCCT GCTATAG TACCCGC ATAGGGC AAGGTCT ACAGGAT
0	TAAGACG CAGTGA GGCCTGC CCTCCAC ATCAACA TGCCACG TTTGTTT CTGGTAC CAGAAGT CGGAACG ATGGAGA
0	AAGTTCT CCAGCCA TCCCAAG AGGTAGG ATTCATC TTTTGCA GTAGGCT CATGGTT CCATGCG AGGATCC CGAACGT
1	ACATACG TCCTCAT ATGCGGG ACGAATT TTGGTAT TGAGCAG AAACGTT GGCGGTT AATTCGT AACGGTA TCTCATA ACGGGCA
0	CGGAAT TATAGGT TATGCGT GATCTCG TTCATAA CATACGG CCTCGGA TACCAAA GTTAATT
0	ACGACCT TTCTTT GTTGGAC TAATAGA ACTGGCC TACTGTA AAATCCC CAGAAGC GCAGGTT GACGGCC
0	TAATTAT AGACA CTCAG CGTCCTG TTTCCGA AGAGACA GATGTAG AGCCTTC CTGTTA AAGGTCC GTCAAAG GACTATG
0	TGCTCCG TAGGTAG TGCTCTT AAGGC CTTGCCC AACCTTG TCCGAGA GACAAGT GAATTTG GTCCCCG TAGGTGT AGCTCT
1	ACTTGGA AGCCCCT GGGCTAT TACCGAG TGCACGA CGGGGAT GGCTGAT GAGCGAG TCTGAGG ACGTTCC CTTCATC TTGCAAG TGTGGCG
1	ACGCTCA CGTTACT ACTTCAG TCGGCCG TGCAATC GCCACAT GTCCAGG TTGTCAC GGTCGAG GACGCCT TTGCGGC AGTTGTT TGGGGAC GTTGTGA
```

Figure 1: Sample gene sequences and their class from sequences.txt

## 3 Logistics

This is an individual project. All handins are electronic. Clarifications and corrections will be announced on Blackboard.

# 4 Handout Instructions

## 4.1 How to start

I Accept the GitHub Classroom assignment using the link: https://classroom.github.com/a/4JlZYxmW

II Clone the GitHub repository created for you to a Linux machine in which you plan to do your work (We advice you to do your work on our linux servers [linuxpool.ku.edu.tr]. See Section 8 for details.)

```
$ git clone https://github.com/COMP-201-Spring-2023/assignment-1-USER.git
```

(Replace USER with your GitHub username that you use to accept the assignment)

III **[IMPORTANT]** After cloning the repository, you are required to write the following honor code in a new file called ”honor.txt” and commit & push it: ”I hereby declare that I have completed this assignment individually, without support from anyone else.” You can use the following command to create ”honor.txt” file and write the honor code in it:

```
$ echo "I hereby declare that I have completed this assignment individually,
without support from anyone else." > honor.txt
```

## 4.2 Main Task

You will write your code in the main.c file. A template file is provided, so that you only have to write the wanted functionality in their respective positions. The positions are indicated with comments.

## 4.3 How to Read a File

- First, open the text file using the `fopen()` function.
- Then, use `fgets()` function to read text from the stream, and store it as a string. The newline of EOF character stops `fgets()` so you can check the newline or EOF file character to read the whole line. **Note:** `fgets()` also reads endline characters, and you should ignore newline character at the end of the sentence.
- When you are done with the file, close the file with the `fclose()` function.
- For more information, you can refer to these links: Link 1, Link 2

# 5 Tasks

## Task 1: Analyze Frequencies

For this task, you will enter a word as second argument where the first argument is a string that determines the task. After entering the word, the code should compute the word's **tf** scores for both healthy and cancerous sequences, so you need to calculate 2 **tf** scores for each word. Then, you should compute which is larger, and print an output about whether the word is mostly in cancerous or healthy sequences.

**tf** score is calculated as:

$$\textbf{tf}(\textbf{word}, \textbf{class}) = \frac{\text{Count of } \textbf{word} \text{ in } \textbf{class} \text{ sequences}}{\text{Count of } \textbf{class} \text{ sequences}}, \ \textbf{class} \in \{\text{healthy, cancerous}\} \tag{1}$$

You can read the above equation as below for word **CATCCTG** and class **healthy** like this:

$$\textbf{tf}(\textbf{CATCCTG}, \textbf{healthy}) = \frac{\text{Count of } \textbf{CATCCTG} \text{ in } \textbf{healthy} \text{ sequences}}{\text{Count of } \textbf{healthy} \text{ sequences}} \tag{2}$$

In Equation 2, we define the **tf** calculation process for **healthy** sequences. But, you can generalize this process to **cancerous** sequences.

First, find how many times the given word occurs in healthy sequences. Then, count the number of healthy sequences of the data set. Then, you can calculate **tf** score for healthy sequences. Also don't forget to use floats for calculations, output resulting float values in 6 precision (use %.6f).

### Sample Outputs

```
$ ./main calculate-tf TAACTAC
Total Healthy:  30000
Word in Healthy:  21
Calculated tf value for this word:  0.000700
Total Cancerous:  30000
Word in Cancerous:  6
Calculated tf value for this word:  0.000200
This word is mostly used for healthy sequences.
$ ./main calculate-tf GGGCTGA
Total Healthy:  30000
Word in Healthy:  22
Calculated tf value for this word:  0.000733
Total Cancerous:  30000
Word in Cancerous:  43
Calculated tf value for this word:  0.001433
This word is mostly used for cancerous sequences.
$ ./main calculate-tf COMP201
This word doesn't occur in the text!
```

## Task 2: Cancerous Gene Sequence Detection

In this task, 8 words will be entered instead of a single word in the previous task. Each word is a separate console argument. You have to calculate **tf-idf** score for each of the words (for both healthy and cancerous class), and sum **tf-idf** score of each word for each class to calculate **tf-idf** score of the sequence for both classes. The sequence belongs to the class with higher **tf-idf** score.

For each word, **tf-idf** score is calculated as **tf** × **idf** scores. The **tf** calculation is same as **Task 1**. The **idf** score is defined as:

$$\textbf{idf}(\textbf{word}) = ln(\frac{\text{Count of all sequences}}{\text{Count of all sequences that include } \textbf{word}}) \tag{3}$$

As you can see, **idf** score does not change between classes. To use $ln$ (natural logarithm) function, `math` library is included in `main.c`, you can use `logf()` function for more accurate floating type calculations.

**CAUTION** Sometimes **word** might not exist in any sequence, and division by zero can happen. You **must** handle this case such as when the denominator of **idf** calculation is zero, you return **idf** zero. Also don't forget to use floats for calculations, output resulting float values in 6 precision (use %.6f).

**Sample Outputs**

Below are some prediction results. In general, the model makes correct predictions. However, as this is a really simple model, it may make wrong predictions, check the sequence that starts with **TAGAGAG**. We will only check whether your **tf-idf** scores are equal to our **tf-idf** scores.

```
   $ tail -3 sequences.txt
1 CTGCTAG GGAAATT TACTTAT TGGAGTG GGGCTGA CGTTTCG AGTCGGG GGTGCCG TCTAACG
CGGCGAT CTGCAGA TGTGCAC
0 TCTAAAG TGATGGG CCCTT GTGGTAT GAATCTT ATCACCG TGGCATA GTCAAAT TTTTAGA GTTGACA
CACGGA TTTGAA
0 TAGAGAG CCGATGG TGTTGAC GACGTGC GATGGA TACCAAA CCCAAAT ATAGTGA GTCTAAA
GAACCTG AGTTACG TTATCAT
$ ./main predict CTGCTAG GGAAATT TACTTAT TGGAGTG GGGCTGA CGTTTCG AGTCGGG
GGTGCCG
Total tf-idf score from healthy messages for the sequence:  0.035137
Total tf-idf score from cancerous sequence for the sequence:  0.064694
This sequence is cancerous.
$ ./main predict TCTAAAG TGATGGG CCCTT GTGGTAT GAATCTT ATCACCG TGGCATA GTCAAAT
Total tf-idf score from healthy messages for the sequence:  0.034271
Total tf-idf score from cancerous sequence for the sequence:  0.038637
This sequence is cancerous.
$ ./main predict TAGAGAG CCGATGG TGTTGAC GACGTGC GATGGA TACCAAA CCCAAAT ATAGTGA
Total tf-idf score from healthy messages for the sequence:  0.033032
Total tf-idf score from cancerous sequence for the sequence:  0.031504
This sequence is not cancerous.
  $ ./main predict COMP201 COMP201 COMP201 COMP201 COMP201 COMP201 COMP201
COMP201
Total tf-idf score from healthy messages for the sequence:  0.000000
Total tf-idf score from cancerous sequence for the sequence:  0.000000
Tf-idf scores are found to be 0, cancerous sequence detection failed!
```

## 6 Evaluation

Your score will be computed out of a maximum of 100 points based on the following distribution:

**35** Task 1.

**55** Task 2.

**5** Style points.

**5** Effective use of version control points.

**Task Points.** Your exact outputs will be matched to some selected test cases (all are different from samples). Therefore, to ease auto grading **you must match your outputs to the outputs in the samples.** You will get points separately from each of the test cases. So, you can get partial points.

To compile your code in **main.c** use the given **Makefile**. To generate the **main** executable, you need to run **make** command. Run your code using ”calculate-tf” and ”predict” arguments for tasks 1 and 2 respectively, just like in the samples.

**Effective Use of Version Control Points.** You are required to push your changes to the repository frequently. If you only push the final version, even if it is implemented 100% correctly, you will lose a fraction of the grade because you are expected to learn to use Version Control Systems effectively. You do not have to push every small piece of change to Github but every meaningful change should be pushed. For example, each of the functions coded and tested can be one commit. **For each task, there should be at least one commit (with proper commit message) that includes just modifications on that task.**

**Style Points.** Finally, we've reserved 5 points for a subjective evaluation of the style of your solutions and your commenting. Your solutions should be as clean and straightforward as possible. Your comments should be informative, but they need not be extensive.

**Important Note:** We use automated plagiarism detection to compare your assignment submission with others and also the code repositories on GitHub and similar sites. Moreover, we plan to ask randomly selected 10% of students to explain their code verbally after the assignments are graded. And one may lose full credit if he or she fails from this oral part.

## 7 Handin Instructions

As with Assignment 0, we use GitHub for the submissions as follows. Note that we want you to get used to using a version management system (Git) in terms of writing good commit messages and frequently committing your work so that you can get most out of Git.

I Commit all the changes you make: `$ git commit -a -m "commit message"`
**Note:** Use meaningful commit messages as in huge projects they become really helpful.
Try to gain this habit from early on.

II Push your work to GitHub servers: `$ git push origin main`

## 8 How to use linuxpool.ku.edu.tr linux servers [1]

I Connect to KU VPN (If you are connected to the KU network, you can skip this step.)
See for details: https://confluence.ku.edu.tr/kuhelp/ithelp/it-services/network-and-wireless/vpn-access

II Connect to linuxpool.ku.edu.tr server using SSH (Replace USER with your Koç University username):

```
$ ssh USER@linuxpool.ku.edu.tr
```

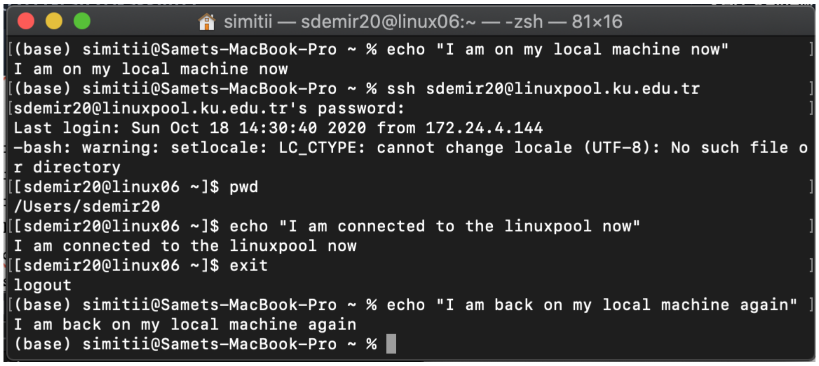
(It will ask your password, type your Koç University password.)

III When you are finished with your work, you can disconnect by typing: `$ exit`

Your connection to the server may drop sometimes. In that case, you need to reconnect.

We advice you to watch the following video about the usage of SSH, which is used to connect remote servers, and SCP, which is used to transfer files between remote servers and your local machine: https://www.youtube.com/watch?v=rm6pewTcSro

[1] For details, please see the guide on linuxpool that we have announced on Blackboard



```
(base) simitii@Samets-MacBook-Pro ~ % echo "I am on my local machine now"
I am on my local machine now
(base) simitii@Samets-MacBook-Pro ~ % ssh sdemir20@linuxpool.ku.edu.tr
sdemir20@linuxpool.ku.edu.tr's password:
Last login: Sun Oct 18 14:30:40 2020 from 172.24.4.144
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file o
r directory
[sdemir20@linux06 ~]$ pwd
/Users/sdemir20
[sdemir20@linux06 ~]$ echo "I am connected to the linuxpool now"
I am connected to the linuxpool now
[sdemir20@linux06 ~]$ exit
logout
(base) simitii@Samets-MacBook-Pro ~ % echo "I am back on my local machine again"
I am back on my local machine again
(base) simitii@Samets-MacBook-Pro ~ %
```

Figure 2: How to connect and disconnect using SSH

## 9 Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else. See Koç University - Student Code of Conduct.

## 10 Late Submission Policy

**You may use up to 7 grace days (in total) over the course of the semester for the assignments.** That is you can submit your solutions without any penalty if you have free grace days left. Any additional unapproved late submission will be punished (1 day late: 20% off, 2 days late: 40% off) and **no submission after 2 days will be accepted.**