TOBB ETU – 2025 Fall
BIL 214 – Exercises 6

Instructions

- Download “Exercises6.zip” to write and test your codes.
- “exercises6.h” includes function declarations. Do not change “exercises6.h”.
- Write function definitions to “exercises6.c”. Do not include any library to “exercises6.c”.
- Directory “inputs” includes input files. You can use “main.c” to test the functions.
- Directory “results” includes the expected output files. “results.txt” stores the expected outputs.
- “exercises6.h” also includes constant definitions to be used in the function prototypes. You can use them when needed. Do **not** assume anything (like the number of lines in a file) that is not specified.
- You can only use Standard C Library Functions that are listed in Worksheet_C.
- Each string argument takes the path of the related file.
- Return 1 (success) at the end of each function. if a file does not open use perror() to display the error massage and return 0 (fail). Do not forget to close opened files before exit.
- You can assume that parameters are always passed correctly, and the lengths of the arrays are sufficient.
- You can assume that the number of characters of a line in a text is always smaller than BUF_LEN.

Questions

soru01. The function compares the contents of two text files. It saves 1 to the integer pointed by “isEqual” if they are identical, 0 otherwise.

soru02. The function copies the text file (source) to another text file (destination) while making lower-case letters upper-case.

soru03. The function saves the number of occurrences of each letter of English alphabet (total of CHAR_NUM) in the text file to another text file. Count the lower-case and upper-case letters together.

soru04. The function saves the number of occurrences of words by length (one-letter words, two-letter words, …, (WORD_LEN-1)-letter words) in the text file to another text file.

soru05. The function copies the text file (source) to another text file (destination) while skipping (not copying) the lines that do not start with ‘a’, ‘b’, ‘c’, ‘d’ or ‘e’.

soru06. The function saves the number of words in each line of the text file to another text file line by line. Assume that words are separated by exactly 1 space.

soru07. The function copies the text file (source) to another text file (destination) while replacing all the occurrences of given word with another (newWord). You can use the function “strcmp”.

Note: For questions 8-10, assume that each line of the input text file is in the following format:
<int (ID)> <string (name)> <int (age)> <float (balance)>

soru08. The function copies customers (each recorded as a line) of the text file (source) whose age is smaller than 25 to another text file (destination) in the following format:
“Account balance of customer <customer_ID> (<customer_name>) is <customer_balance>\n”

soru09. The function appends information of new customer (ID, name, age, balance) to the end of the text file. You can use mode “a”.

soru10. The function increases the “balance” of the customer with given “ID” by the amount of “change” (can be negative of 0) by opening the text file with “r+” mode (Open the file only once.). Hint: Use rewind.

Note: Use the struct below (defined in exercises4.h) for questions 11-15.

```
typedef struct {
        unsigned int ID, age;
        char name[16];
        float balance;
} Customer;
```

soru11. The function copies the binary file (source) to another binary file (destination).

soru12. The function displays information related to the i-th customer (starts from 1) in the given binary file. If reading fails, display nothing. Time and space complexity of the function must be O(1).

soru13. The function updates the information related to the i-th customer (starts from 1) in the given binary file. If “i” is 0, add the customer data to the end of the binary file. Time and space complexity of the function must be O(1).

soru14. The function increases the “balance” of i-the customer (starts from 1) in the given binary file by the amount of “change” (can be negative of 0). Time and space complexity of the function must be O(1).

soru15. The function returns the average (avg) and maximum (max) of customer balances in the given binary file while reading from the file “N” items at a time. Note that the number of last read items can be less than N. Space complexity of the function must be O(N) and time complexity of the function must be O(M) where M is the number of items.