PROBLEMS

# 1.

Write a program that for given textual file text.txt will find the ratio of vowel/consonants.

# 2.

Write a program that each row from given input file input.txt will copy in other file output.txt, so that for each row from input.txt will add one more with the length of that row. Each row can have at most 80 characters.

# 3.

Write a program that will read elements of a matrix written in text file with name  
mat1.txt. In the first line of the file are written the number of rows and columns of  
the matrix. Each element of the matrix is floating point number written in separate  
line. The transposed matrix write in a new output file mat2.txt using the same  
format.

# 4.

Given is the file SP\_example.txt. Write a program that will read the file and will print the number of rows with more than 10 vowels, and the total vowels in the file.

# 5.

Write a program that for given file words.txt will print all the words that have three or more equal letters (some letter occurs three or more times). The comparison of letters is not case sensitive. At the end it should print the count of words that satisfy this condition. The file contains one word per row. Each word is composed only from letters. The maximal length of a word is 20 chars.

# 6.

Write a program that will print the count of occurrences of a word composed only from digits (read from SI) in a file named dat.txt.

# 7.

Write a function double relativeFrequency (char \* fileName, char letter) that will determine the relative frequency of a letter in a text file named 'fileName'!.

A relative frequency of a letter is calculated as the quotient between the count of occurrences of the letter and the total number of letters in the text. The case of the letter should be ignored (consider all letters as lower or upper letters!)

After that, write a function void printRelativeFrequencies(char \* fileName) that will utilize the previous function, and will calculate the relative frequencies of all the letters from the alphabet and then it will print them on standard output.

SOLUTIONS

# 1.

*/\*Write a program that for given textual file text.txt will find the ratio of vowel/consonants.\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
int** isVowel(**char** c){  
 **return** tolower(c)==**'a'** || tolower(c)==**'e'** || tolower(c)==**'i'** || tolower(c)==**'o'** || tolower(c)==**'u'**;  
  
}  
  
**int** main(){  
 FILE \*fp= fopen(**"C:\\Users\\user\\Desktop\\CCC\\text.txt"**,**"r"**);  
 **char** c;  
 **int** vowels=0,consonants=0;  
 **while**((c= fgetc(fp))!=**EOF**){  
 **if**(isalpha(c)){  
 **if**(isVowel(c)){  
 vowels++;  
 } **else**{  
 consonants++;  
 }  
 }  
 }  
 fclose(fp);  
 printf(**"Ratio vowels/consonants: %d/%d = %.2f"**,vowels,consonants,(**float**)vowels/consonants);  
**return** 0;

}

# 2.

*/\*Write a program that each row from given input file input.txt will copy in other file  
output.txt, so that for each row from input.txt will add one more with the length  
of that row. Each row can have at most 80 characters.\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
void** clearStr(**char** \*str){  
 **if**(str[strlen(str)-1]==**'\n'**){  
 str[strlen(str)-1]=**'\0'**;  
 }  
}  
  
**int** main(){  
 FILE \*inputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\input.txt"**,**"r"**);  
 FILE \*outputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\output.txt"**,**"w"**);  
 **char** line[80];  
 **while** (fgets(line,**sizeof**(line),inputFile)!=**NULL**){  
 clearStr(line);  
 **int** length= strlen(line);  
 fprintf(outputFile,**"%d\n%s\n"**,length, line);  
 }  
  
 fclose(inputFile);  
 fclose(outputFile);

**return** 0;  
  
}

# 3.

*/\*Write a program that will read elements of a matrix written in text file with name  
mat1.txt. In the first line of the file are written the number of rows and columns of  
the matrix. Each element of the matrix is floating point number written in separate  
line. The transposed matrix write in a new output file mat2.txt using the same  
format.\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
  
  
int** main(){  
 FILE \*inputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\mat1.txt"**,**"r"**);  
 FILE \*outputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\mat2.txt"**,**"w"**);  
 **int** mat[100][100];  
 **int** m,n;  
 fscanf(inputFile,**"%d%d"**,&m,&n);  
 **for**(**int** i=0 ; i<m ; i++){  
 **for**(**int** j=0 ; j<n ; j++){  
 fscanf(inputFile,**"%d"**,&mat[i][j]);  
 }  
 }  
 **int** transposed[100][100];  
  
 **for**(**int** i=0 ; i<n ; i++){  
 **for**(**int** j=0 ; j<m ; j++){  
 transposed[i][j]=mat[j][i];  
 }  
 }  
 **for**(**int** i=0 ; i<n ; i++){  
 **for**(**int** j=0 ; j<m ; j++){  
 fprintf(outputFile,**"%d "**,transposed[i][j]);  
 }  
 fprintf(outputFile,**"\n"**);  
 }  
  
 fclose(inputFile);  
 fclose(outputFile);

**return** 0;  
  
}

# 4.

*/\*Given is the file SP\_example.txt. Write a program that will read the file and will  
print the number of rows with more than 10 vowels, and the total vowels in the file.\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
int** countOfVowels(**char** \*str){  
 **int** countVowels=0;  
 **for**(**int** i=0 ; i< strlen(str) ; i++){  
 **if**(tolower(str[i])==**'a'** || tolower(str[i])==**'e'** || tolower(str[i])==**'i'** || tolower(str[i])==**'o'** || tolower(str[i])==**'u'**){  
 countVowels++;  
 }  
 }  
 **return** countVowels;  
}  
  
**int** main(){  
 FILE \*inputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\SP\_example.txt"**,**"r"**);  
 **char** line[100];  
 **int** totalVowels=0;  
 **int** counter=0;  
 **while**(fgets(line,**sizeof**(line),inputFile)!=**NULL**){  
 **int** countOfVowles= countOfVowels(line);  
 **if**(countOfVowles>10){  
 counter++;  
 }  
 totalVowels+=countOfVowles;  
 }  
 fclose(inputFile);  
 printf(**"Total %d rows have more than 10 vowels.\nThe file has total %d vowels."**,counter,totalVowels);  
 **return** 0;  
}

# 5.

*/\*Write a program that for given file words.txt will print all the words that have three  
or more equal letters (some letter occurs three or more times). The comparison of  
letters is not case sensitive. At the end it should print the count of words that satisfy  
this condition.  
The file contains one word per row. Each word is composed only from letters. The  
maximal length of a word is 20 chars.  
\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
int** wordsWithEqualLetters(**char** \*str){  
 **int** countOfEqualLetters=0;  
 **for**(**int** i=0 ; i< strlen(str) ; i++){  
 **for**(**int** j=i+1 ; j< strlen(str) ; j++){  
 **if**(tolower(str[i])==tolower(str[j])){  
 countOfEqualLetters++;  
 }  
 }  
 }  
 **return** countOfEqualLetters;  
}  
  
**void** clearStr(**char** \*str){  
 **if**(str[strlen(str)-1]==**'\n'**){  
 str[strlen(str)-1]=**'\0'**;  
 }  
}  
  
**int** main(){  
 FILE \*inputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\words.txt"**,**"r"**);  
 **char** word[20];  
 **int** countOfWords=0;  
 **while**(fgets(word,**sizeof**(word),inputFile)!=**NULL**){  
 clearStr(word);  
 **int** countOfEqualLetters= wordsWithEqualLetters(word);  
 **if**(countOfEqualLetters>=3){  
 puts(word);  
 countOfWords++;  
 }  
 }  
 fclose(inputFile);  
 printf(**"Total %d words"**,countOfWords);  
 **return** 0;  
}

# 6.

*/\*Write a program that will print the count of occurrences of a word composed  
only from digits (read from SI) in a file named dat.txt.\*/*#include **<stdio.h>**#include **<string.h>**#include **<ctype.h>  
  
int** countOccurrencesOfWord(**char** \*str, **char** \*subStr){  
 **int** countOcc=0;  
 **char** \*res= strstr(str,subStr);  
 **while**(res!=**NULL**){  
 countOcc++;  
 res= strstr(res+strlen(subStr),subStr);  
 }  
 **return** countOcc;  
}  
  
**int** countOccurrencesOfWordR(**char** \*str, **char** \*subStr){  
 **char** \*res= strstr(str,subStr);  
 **if**(res==**NULL**){  
 **return** 0;  
 } **else**{  
 **return** 1+ countOccurrencesOfWordR(res+ strlen(subStr),subStr);  
 }  
}  
  
**int** main(){  
 FILE \*inputFile= fopen(**"C:\\Users\\user\\Desktop\\CCC\\dat.txt"**,**"r"**);  
 **char** word[100];  
 scanf(**"%s"**,word);  
 **char** line[100];  
 **int** counter=0;  
 **while**(fgets(line,**sizeof**(line),inputFile)!=**NULL**){  
 counter+= countOccurrencesOfWord(line,word);  
 }  
 printf(**"%d"**,counter);  
 fclose(inputFile);  
 **return** 0;  
}

# 7.

*/\*Write a function double relativeFrequency (char \* fileName, char letter) that will determine the relative frequency  
 of a letter in a text file named 'fileName'!.  
A relative frequency of a letter is calculated as the quotient between the count of occurrences  
of the letter and the total number of letters in the text. The case of the letter should be ignored  
(consider all letters as lower or upper letters!) After that, write a function void printRelativeFrequencies(char \* fileName) that will utilize the  
previous function, and will calculate the relative frequencies of all the letters from the alphabet and  
then it will print them on standard output.  
\*/*#include **<stdio.h>**#include **<ctype.h>  
  
double** relativeFrequency (**char** \* fileName, **char** letter){  
 FILE \*f= fopen(fileName,**"r"**);  
 **int** totalLetters=0, countL=0;  
 **char** c;  
 **while**((c=fgetc(f))!=**EOF**){  
 **if**(isalpha(c)){  
 totalLetters++;  
 }  
 **if**(tolower(letter)== tolower(c)){  
 countL++;  
 }  
 }  
 **return** (countL\*1.0)/totalLetters;  
}  
  
**void** printRelativeFrequencies(**char** \* fileName){  
 **for**(**char** c=**'A'** ; c<=**'Z'** ; c++){  
 printf(**"%c -> %.3lf\n"**,c, relativeFrequency(fileName,c));  
 }  
}  
  
**int** main(){  
 printRelativeFrequencies(**"C:\\Users\\user\\Desktop\\CCC\\Exams\\input.txt"**);  
 **return** 0;  
}