//Cathal Lawlor - 21325456

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#include <stdbool.h>

char alltext[200000];

char oneline[1000];

const char\* delimeters = " \n\t,";

int wordSyllables(char\* word);

int main() {

FILE\* file\_ptr;

//file\_ptr = fopen("article-irish-times.txt", "r"); // open for reading

file\_ptr = fopen("article-green-eggs-and-ham.txt", "r");

if (file\_ptr == NULL) {

printf("Could not open article\n");

}

else {

alltext[0] = '\0'; // to make sure the string is empty

while (fgets(oneline, 999, file\_ptr) != NULL) {

// read the next line and append it (keeping /n intact)

strcat(alltext, oneline);

}

fclose(file\_ptr);

// loop the whole text to count sentences

// note that we do this before running strtok since strtok modifies the string itself

int sentences = 0;

int len = strlen(alltext);

for (int i = 0; i < len; i++) {

char c = alltext[i];

if (c == '.' || c == ':' || c == ';' || c == '?' || c == '!') {

sentences++; }

}

// Calculate the Flesch Readability Index

char\* word = strtok(alltext, delimeters);

int syllables = 0;

int words = 0;

while (word != NULL) {

// loop each word to count them and their syllables

words++;

syllables += wordSyllables(word);

word = strtok(NULL, delimeters);

}

float flesch = 206.835 - 84.6 \* ((float)syllables / words) - 1.015 \* ((float)words / sentences);

printf("Words=%d Syllables=%d Sentences=%d\n", words, syllables, sentences);

printf("Flesch Readability Index = %f\n", flesch);

}

}

int wordSyllables(char\* word) {

// Each group of adjacent vowels (a, e, i, o, u, y) counts as one syllable

// However excluding "e" at the end of a word as it doesn't count as a syllable

int s = 0;

bool inVowelGroup = false;

int len = strlen(word);

for (int i = 0; i < len; i++) {

char c = tolower(word[i]);

bool isVowel = (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' || c == 'y');

if (isVowel && !inVowelGroup) {

if (i + 1 < len || c != 'e')

s++;

}

inVowelGroup = isVowel;

}

if (s == 0) // 0-syllable words not allowed

return 1;

return s;

}