#include <stdio.h>

#include <string.h>

#include <malloc.h>

#include <iostream>

void main(void) {

FILE \*fpin; // входной файл

char \* buffer = NULL;

char \*\*lines = NULL; // указатель на динамический массив

char time;

int i = 0;

int lsize = 0;

int k = 0;

fpin = fopen("test.txt", "rt"); // открыть файл для чтения

while (!feof(fpin))// цикл до конца входного файла

{

time = fgetc(fpin); // считываем очередной элемент текста

lsize++; // счетчик длины предложения чтобы выделить нужное количество памяти в массиве, а также, чтобы передвинуть указатель в файле на нужное место, чтобы считать лишь нужную часть предложения

i++; // используется для отслеживания места считывания, т.е "указатель" на очередной символ, который будет считан

if (time == '.')

{

k++; //счетчик нужного количества строк

//HI.

fseek(fpin, i - lsize, SEEK\_SET); //перемещение на АКТУАЛЬНОЕ СЧИТЫВАНИЕ - КОЛИЧЕСТВО БУКВ = НАЧАЛО НОВОГО ПРЕДЛОЖЕНИЯ

buffer = (char\*)realloc(buffer,(lsize+1)\*sizeof(char) ); //выделение нужного количества памяти в массиве BUFFER

fread(buffer, 1, lsize, fpin); //считываем массив BUFFER

buffer[lsize] = '\0';

// заказать динамический массив указателей c выделением памяти

lines = (char\*\*)realloc(lines, k \* sizeof(char\*));

// заказать в памяти место для строки плюс один байт

lines[k - 1] = (char\*)malloc(sizeof(char) \*(lsize + 1));

//копируем. Очищаем буфер

strcpy(lines[k - 1], buffer);

fseek(fpin, 1, SEEK\_CUR); // перемещение на +1, чтобы миновать точку

i++;

lsize = 0; // После каждой точки идет пробел, его считать не нужно, поэтому -1, а не 0

}

}

fclose(fpin);

for (int i = 0; i<k; i++)

{

puts(lines[i]);

free(lines[i]);

lines[i] = NULL;

}

free(lines);

lines = NULL;

free(buffer);

buffer = NULL;

}