```
#include "Terminal_IO.h"
#include <sys/mman.h>
#include <sys/stat.h>
#include <unistd.h>
#include <fcntl.h>

//void printHistogram(Histogram *h, int index);

/**
 * Druckt ein Histogram im Terminal aus.
 */
void printHistogramStruct(Histogram **h, int index) {
    int n;
    // Drucke Cursor
    printf("h[%d].cursor = %d\n", index, (*h[index]).cursor);

    for (n = 0; n < 52; n++) {
        if (n >= 0 && n < 26) { // GroÃM−^_buchstaben
            printf("h[%d].letter[%c] = %d\n", index, (char) n+65, (*h[index]).letter[n
]);
        }

        if (n >= 26 && n < 52) {
            printf("h[%d].letter[%c] = %d\n", index, (char) n−26+97, (*h[index]).lette
r[n]);
        }
        //printf("h[%d].letter[%d] = %d\n", index, n, h[index].letter[n]);
    }
}



/**
 * Druckt alle Elemente des Histograms (Letter Array und Cursor) auf dem Termina
l aus.
 *Â @param h Histogramme
 * @param size Anzahl der Histogramme
 */
void printHistogramArray(Histogram **h, int size) {
  //printf("call HistogramArray(), size = %d\n", size);
    int index;
    char *zeile;
    for (index = 0; index < size; index++) {
        zeile = getHistogramAsString(h[index]);
        printf("%d: %s\n", index, zeile);
    }
}

void _printHistogramArray(Histogram *data, int size) {
    int index;
    char *zeile;
    for (index = 0; index < size; index++) {
        zeile = getHistogramAsString(data+index);
        printf("%d: %s\n", index, zeile);
    }
}

void printControlLines(Histogram **h, const char *file_in, int line){
    FILE *datei;
    datei = fopen(file_in, "r");

    unsigned int cursor;
```



```
  //fseek(datei, 0L, SEEK_END);
  //unsigned int length = ftell(datei);

    cursor = (*h[(line − 1)]).cursor;
    fseek(datei, cursor, SEEK_SET);

    char c;
    printf("Line %d: ", line);
    while( (c = fgetc(datei)) != '\n'){
        printf ("%c", c);
    }
    printf ("\n \n");
}
```