

## PRAKTIKUM POINTER-STARTTER

INPUT & PENJELASAN :

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4
5  //untuk panjang maksimum dan minimum teks.
6  #define MAX_LENGTH 2024
7  #define MIN_LENGTH 2023
8
9  // menerima pointer ke integer sebagai parameter
10 void lessThanRequired(int *length);
11 void equalThanRequired();
12 void moreThanRequired(int *length);
13
14 int checkLengthRequirement(char* text); //menerima array karakter sebagai parameter.
15 //tidak ada yang di ubah(hanya di pindahkan)
16 int main() {
17     int lengthOfText, selectOption;
18     FILE *fptr = NULL;
19     char text[MAX_LENGTH];
20
21     // Membuka file untuk dibaca
22     fptr = fopen("file.txt", "r");
23
24     // Memeriksa apakah file berhasil dibuka
25     if(fptr == NULL){
26         printf("Error");
27         exit(1);
28     }
29
30     // Membaca teks dari file
31     fgets(text, MAX_LENGTH, fptr);
32
33     // Menutup file setelah selesai membaca
34     fclose(fptr);
35
36     // Memeriksa panjang teks dan menentukan tindakan yang tepat
37     selectOption = checkLengthRequirement(text);
38
39     // Menghitung panjang teks menggunakan strlen
40     lengthOfText = strlen(text);
41
42     // Memanggil fungsi berdasarkan nilai selectOption
43     void (*functions[3])(int*) = {lessThanRequired, equalThanRequired, moreThanRequired};
44     functions[selectOption](&lengthOfText);
45
46     printf("\nThe Length is updated to %d", lengthOfText); // Menampilkan panjang teks setelah diupdate
47
48     return 0;
49 }
50
51 // Fungsi utama dari program.
52 // Membuka file "file.txt" untuk dibaca.
53 // Membaca teks dari file ke dalam array text.
54 // Menutup file setelah selesai membaca.
55 // Memanggil checkLengthRequirement untuk menentukan tindakan yang tepat berdasarkan panjang teks yang dibaca.
56 // Menghitung panjang teks menggunakan fungsi strlen().
57 // Memanggil fungsi yang sesuai berdasarkan nilai yang dikembalikan oleh checkLengthRequirement.
58 // Menampilkan panjang teks setelah diperbarui.
59
60 //fungsi yang dipanggil untuk menangani tindakan yang sesuai berdasarkan panjang teks.
61 void lessThanRequired(int *length) {
62     printf("\nThe length of your text is less than specified, please update your text");
63     printf("\nLength Before : %d", *length);
64     *length = MIN_LENGTH;
65 }
66
67 void equalThanRequired() {
68     printf("\nThank you, Your text length is correct");
69 }
70
71 void moreThanRequired(int *length) {
72     printf("\nYour text is too long, please reduce the text");
73     *length = MIN_LENGTH;
74 }
75
76 // untuk menentukan apakah panjang teks sesuai dengan kebutuhan.
77 int checkLengthRequirement(char* text){
78     int length = strlen(text);
79     return (length < MIN_LENGTH) ? 0 : ((length == MIN_LENGTH) ? 1 : 2);
80 }
81
```

OUTPUT :

```
PS C:\Users\david louis> cd "c:\Users\david louis\OneDrive\Documents\ASD\ASD\" ;  
The length of your text is less than specified, please update your text  
Length Before : 312  
The Lenght is updated to 2023
```