

Ilham prakosa

1203230027

LAPRAK ASD

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 #define MAX_LENGTH 10
6 #define MIN_LENGTH 7
7
8 void lessThanRequired (int* lengthOfText){
9     printf("The length of your text is less than specified, please update your text\n");
10    *lengthOfText = MIN_LENGTH;
11 }
12
13 void equalThanRequired (){
14     printf("Thank you, Your text length is correct\n");
15 }
16
17 void moreThanRequired (int* lengthOfText){
18     printf("Your text is too long, please reduce the text\n");
19     *lengthOfText = MIN_LENGTH;
20 }
21
22 int checkLenghtRequirement(char* text){
23     int length = strlen(text);
24     if (length < MIN_LENGTH)
25         return 0;
26     else if (length == MIN_LENGTH)
27         return 1;
28     else
29         return 2;
30 }
31
32 int main() {
33     int lengthOfText, selectOption;
34     FILE *fptr = NULL;
35     char text[MAX_LENGTH];
36
37     fptr = fopen("file.txt", "r");
38
39     if(fptr == NULL){
40         printf("Error");
41         exit(1);
42     }
43
44     fgets(text, MAX_LENGTH, fptr);
45
46     fclose(fptr);
47
48     selectOption = checkLenghtRequirement(text);
49
50     // TODO: Use function pointers to call the appropriate function based on selectOption
51     // HINT: Create an array of function pointers and call the appropriate function using selectOption
52     void (*functionArray[3])(int*) = {lessThanRequired, equalThanRequired, moreThanRequired};
53     functionArray[selectOption](&lengthOfText);
54
55     printf("\nThe Length is updated to %d", lengthOfText);
56
57     return 0;
58 }
```

Mari kita jelaskan setiap bagian dari kode tersebut:

Libraries (`#include <stdio.h>`, `#include <stdlib.h>`, `#include <string.h>`)

Baris-baris ini menyertakan header file standar yang diperlukan untuk operasi input/output, alokasi memori, dan manipulasi string.

```
### Konstanta (`#define MAX_LENGTH 10`, `#define MIN_LENGTH 7`)
```

Ini mendefinisikan dua konstanta: `MAX_LENGTH` dan `MIN_LENGTH`. `MAX_LENGTH` menentukan panjang maksimum yang dapat diinputkan, sedangkan `MIN_LENGTH` menunjukkan panjang minimum yang diperlukan.

```
### Fungsi-fungsi:
```

```
1. `void lessThanRequired(int* lengthOfText)`
```

- Fungsi ini mencetak pesan kesalahan jika panjang teks kurang dari panjang minimum yang ditentukan. Kemudian, panjang teks diubah menjadi panjang minimum.

```
2. `void equalThanRequired()`
```

- Fungsi ini mencetak pesan apabila panjang teks sama dengan panjang minimum yang ditentukan.

```
3. `void moreThanRequired(int* lengthOfText)`
```

- Fungsi ini mencetak pesan kesalahan jika panjang teks lebih panjang dari panjang minimum yang ditentukan. Kemudian, panjang teks diubah menjadi panjang minimum.

```
4. `int checkLengthRequirement(char* text)`
```

- Fungsi ini mengembalikan nilai berdasarkan panjang teks yang diberikan:

- `0` jika panjang teks kurang dari panjang minimum,
- `1` jika panjang teks sama dengan panjang minimum, dan
- `2` jika panjang teks lebih panjang dari panjang minimum.

```
### Fungsi `main()`
```

1. Mendeklarasikan variabel-variabel:

- `int lengthOfText`: Menyimpan panjang teks.
- `int selectOption`: Menyimpan hasil dari pengecekan panjang teks.
- `FILE *fptr`: Pointer untuk file.
- `char text[MAX_LENGTH]`: Array untuk menyimpan teks.

2. Membuka file `"file.txt"` dalam mode baca `"r"`.

- Jika gagal, cetak "Error" dan keluar dari program.

3. Membaca teks dari file menggunakan `fgets()` dan menyimpannya dalam array `text`.

4. Menutup file.

5. Memeriksa panjang teks menggunakan fungsi `checkLengthRequirement()` dan menyimpan hasilnya dalam `selectOption`.

6. Membuat array of function pointers `functionArray` yang menunjuk ke fungsi-fungsi yang sesuai.

7. Memanggil fungsi yang sesuai dengan `selectOption` menggunakan `functionArray` dan melewati `lengthOfText` sebagai argumen.

8. Mencetak panjang teks yang diperbarui.

9. Mengembalikan `0` sebagai penanda bahwa program selesai dijalankan.

Input teks:

```
1  manusia
```

Output teks:

```
Thank you, Your text length is correct
```

Input teks:

```
1 manusiabiasa
```

Output:

```
Your text is too long, please reduce the text
```

```
The Length is updated to 7
```

Input teks:

```
1 manus
```

Output teks:

```
The length of your text is less than specified, please update your text
```

```
The Length is updated to 7
```