

Nama : I Made Suta Eka Dharma

Kelas : IF 03-01

NIM : 1203230072

LATIHAN SOAL ALGORITMA DAN STRUKTUR DATA

SOURCE CODE

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MAX_LENGTH 2024
#define MIN_LENGTH 1945

void lessThanRequired (int *length){
    printf("The length of your text is less than specified, please update your text\n");
    printf("Length Before : %d", *length);
    *length = MAX_LENGTH;
}

void equalThanRequired (int *length){
    printf("Thank you, Your text length is correct\n");
}

void moreThanRequired (int *length){
    printf("Your text is too long, please reduce the text\n");
    printf("Length Before : %d", *length);
    *length = MAX_LENGTH;
}

int checkLengthRequirement(char* text){
    int length = strlen(text);
    if (length < MIN_LENGTH)
        return 0;
    else if (length == MIN_LENGTH)
        return 1;
    else
        return 2;
}

int main() {
    int lengthOfText, selectOption;
    FILE *fptr = NULL;
    char text[MAX_LENGTH];
```

```

    fptr = fopen("C:\\Users\\Asus\\Documents\\tugas praktikum ASD\\file.txt",
"r");

    if(fptr == NULL){
        printf("Error");
        exit(1);
    }

    fgets(text, MAX_LENGTH, fptr);

    fclose(fptr);
    lengthOfText = strlen(text);
    selectOption = checkLengthRequirement(text);

    void (*function[3])(int*) = {lessThanRequired, equalThanRequired,
moreThanRequired};
    function[selectOption](&lengthOfText);

    printf("\nThe Length is updated to %d", lengthOfText);

    return 0;
}

```

OUTPUT

```

PS C:\Users\Asus\vscode> cd "c:\Users\Asus\vscode\eko\" ; if ($?) { gcc contoh.c -o contoh } ; if ($?) { .\contoh }
The length of your text is less than specified, please update your text
Length Before : 312
The Length is updated to 2024
PS C:\Users\Asus\vscode\eko>

```

PENJELASAN SOURCE CODE :

```

#define MAX_LENGTH 2024
#define MIN_LENGTH 1945

```

- Berfungsi untuk mendeklarasi Panjang minimum dan maksimum text.

```
int checkLengthRequirement(char* text){
    int length = strlen(text);
    if (length < MIN_LENGTH)
        return 0;
    else if (length == MIN_LENGTH)
        return 1;
    else
        return 2;
}
```

- `int checkLengthRequirement (char* text)`;- Baris ini mendeklarasikan fungsi `checkLengthRequirement` dengan parameter `text` yang merupakan penunjuk ke karakter (yaitu string).
- `int length = strlen(text)`;- Baris ini mendeklarasikan variabel integer `length` dan menginisiasinya dengan panjang string `text` menggunakan `strlen` fungsi.
- `if (length < MIN_LENGTH)`- Baris ini memeriksa apakah panjang string `text` kurang dari bilangan bulat konstan `MIN_LENGTH`. Jika ya, maka fungsinya akan kembali 0
- `else if (length == MIN_LENGTH)`- Baris ini memeriksa apakah panjang string `text` sama dengan bilangan bulat konstan `MIN_LENGTH`. Jika ya, maka fungsinya akan berlanjut ke baris berikutnya.

```
void lessThanRequired (int *length){
    printf("The length of your text is less than specified, please update your text\n");
    printf("Length Before : %d", *length);
    *length = MAX_LENGTH;
}

void equalThanRequired (int *length){
    printf("Thank you, Your text length is correct\n");
}

void moreThanRequired (int *length){
    printf("Your text is too long, please reduce the text\n");
    printf("Length Before : %d", *length);
    *length = MAX_LENGTH;
}
```

- Fungsi `lessThanRequired` digunakan ketika panjang teks kurang dari nilai yang dibutuhkan. Ini mencetak pesan yang menunjukkan bahwa panjang teks kurang dari yang ditentukan dan perlu diperbarui. Kemudian ia mengatur panjang teks ke panjang maksimum yang diperbolehkan.
- Fungsi `equalThanRequired` tidak diterapkan dalam konteks yang disediakan, tetapi diasumsikan digunakan ketika panjang teks sama dengan nilai yang diperlukan. Ini mencetak pesan yang menunjukkan bahwa panjang teks sudah benar.
- Fungsi `moreThanRequired` digunakan ketika panjang teks lebih besar dari nilai yang dibutuhkan. Ini mencetak pesan yang menunjukkan bahwa teksnya terlalu panjang dan perlu diperkecil. Kemudian ia mengatur panjang teks ke panjang maksimum yang diperbolehkan.

```

int main() {
    int lengthOfText, selectOption;
    FILE *fptr = NULL;
    char text[MAX_LENGTH];
    fptr = fopen("C:\\Users\\Asus\\Documents\\tugas praktikum ASD\\file.txt", "r");

    if(fptr == NULL){
        printf("Error");
        exit(1);
    }
}

```

- FILE *fptr = NULL;- Pointer ke suatu FILE objek dideklarasikan sebagai fptr dan diinisialisasi ke NULL.
- char text [MAX_LENGTH];- Array karakter text dengan ukuran MAX_LENGTH dideklarasikan.
- fptr = fopen("C:\\Users\\Asus\\Documents\\tugas praktikum ASD\\file.txt", "r");- fopen Fungsi ini digunakan untuk membuka file yang terletak di jalur tertentu dalam mode baca dan penunjuk file yang dikembalikan ditugaskan ke fptr.
- if(fptr == NULL) {- Baris ini memeriksa apakah file berhasil dibuka. Jika file tidak dapat dibuka, program akan mencetak "Error" dan keluar dengan kode status 1.