

PENJELASAN TUGAS CODINGAN ASD POINTER

```
8 void lessThanRequired (int *lengthOfText){
9     printf("\nThe length of your text is less than specified, please update your text");
10     printf("\nLength Before : %d", *lengthOfText);
11     *lengthOfText = MIN_LENGTH;
12 }
```

Fungsi `lessThanRequired()` dipanggil saat panjang teks yang diberikan kurang dari nilai `MIN_LENGTH` yang diharapkan. Fungsi ini bertugas memberikan respons yang sesuai terhadap kondisi tersebut. Ketika dipanggil, fungsi `lessThanRequired()` akan mencetak pesan kesalahan yang menunjukkan bahwa panjang teks yang diberikan tidak mencapai panjang minimum yang diharapkan. Selanjutnya, fungsi ini akan memperbarui nilai `lengthOfText` agar sama dengan nilai `MIN_LENGTH` yang telah ditetapkan sebelumnya. Dengan melakukan operasi aritmatika, fungsi ini menjamin bahwa panjang teks yang diberikan akan memenuhi atau melebihi panjang minimum yang diharapkan. Jadi, implementasi dari fungsi `lessThanRequired()` adalah memberikan pesan kesalahan yang jelas kepada pengguna dan memastikan bahwa panjang teks diperbarui agar memenuhi persyaratan minimum yang telah ditetapkan.

```
14 void equalThanRequired (int *lengthOfText){
15     printf("\nThank you, Your text length is correct");
16 }
```

Ketika fungsi `equalThanRequired()` dipanggil, itu menunjukkan bahwa panjang teks yang dibaca sama dengan nilai minimum yang diharapkan (`MIN_LENGTH`). Fungsi ini bertugas untuk memberikan respons terhadap kasus ini. Dalam hal ini, responsnya adalah mencetak pesan ke layar yang menyatakan bahwa panjang teks sudah sesuai dengan yang diharapkan. Pesan ini memberitahu pengguna bahwa tidak ada masalah dengan panjang teks yang dibaca, karena panjangnya telah mencapai nilai minimum yang ditetapkan sebelumnya. Jadi, implementasi dari `equalThanRequired()` adalah memberikan respons yang jelas dan positif kepada pengguna tentang panjang teks yang telah sesuai dengan standar yang diharapkan.

```
18 void moreThanRequired (int *lengthOfText){
19     printf("\nYour text is too long, please reduce the text");
20     printf("\nThe length is updated to %d", *lengthOfText);
21     *lengthOfText = MAX_LENGTH;
22 }
```

Ketika dipanggil, fungsi `moreThanRequired()` akan mencetak pesan kesalahan yang mengindikasikan bahwa panjang teks terlalu panjang. Pesan ini memberitahu pengguna bahwa teks yang dimasukkan melampaui batas yang diizinkan. Selanjutnya, fungsi ini akan melakukan pembaruan nilai `lengthOfText` menggunakan operasi aritmatika. Panjang teks akan diatur kembali ke nilai maksimum yang telah ditetapkan (`MAX_LENGTH`), sehingga memastikan bahwa panjang teks tidak melebihi batas yang diizinkan. Jadi, implementasi dari `moreThanRequired()` adalah memberikan pesan kesalahan yang jelas kepada pengguna tentang panjang teks yang terlalu panjang, dan melakukan penyesuaian nilai panjang teks ke nilai maksimum yang diizinkan.

```

50     selectOption = checkLenghtRequirement(text);
51     void (*functions[3])(int *) = {lessThanRequired, equalThanRequired, moreThanRequired};
52     lengthOfText = strlen(text);
53     (*functions[selectOption])(&lengthOfText);

```

Dalam main(), hasil yang dikembalikan oleh checkLenghtRequirement() disimpan dalam variabel selectOption. Ini menentukan status panjang teks yang dibaca: apakah kurang dari, sama dengan, atau lebih dari panjang yang diharapkan. Selanjutnya, program menggunakan array pointer ke fungsi untuk memanggil fungsi yang sesuai dengan nilai selectOption. Dengan menggunakan array pointer, program secara dinamis memilih fungsi yang akan dipanggil berdasarkan hasil evaluasi panjang teks, tanpa perlu menggunakan struktur kontrol seperti if atau switch. Setelah pemanggilan fungsi, nilai panjang teks yang baru diperbarui disimpan dalam variabel lengthOfText. Ini memastikan bahwa panjang teks sesuai dengan kebutuhan setelah diproses oleh fungsi-fungsi yang sesuai. Terakhir, program mencetak pesan yang menunjukkan bahwa panjang teks telah diperbarui. Dengan demikian, kode memenuhi persyaratan yang diberikan dalam TODO tanpa menggunakan struktur kontrol, dan juga sesuai dengan panjang kode yang ditentukan.

WICIPTO SETIADI/1203230042