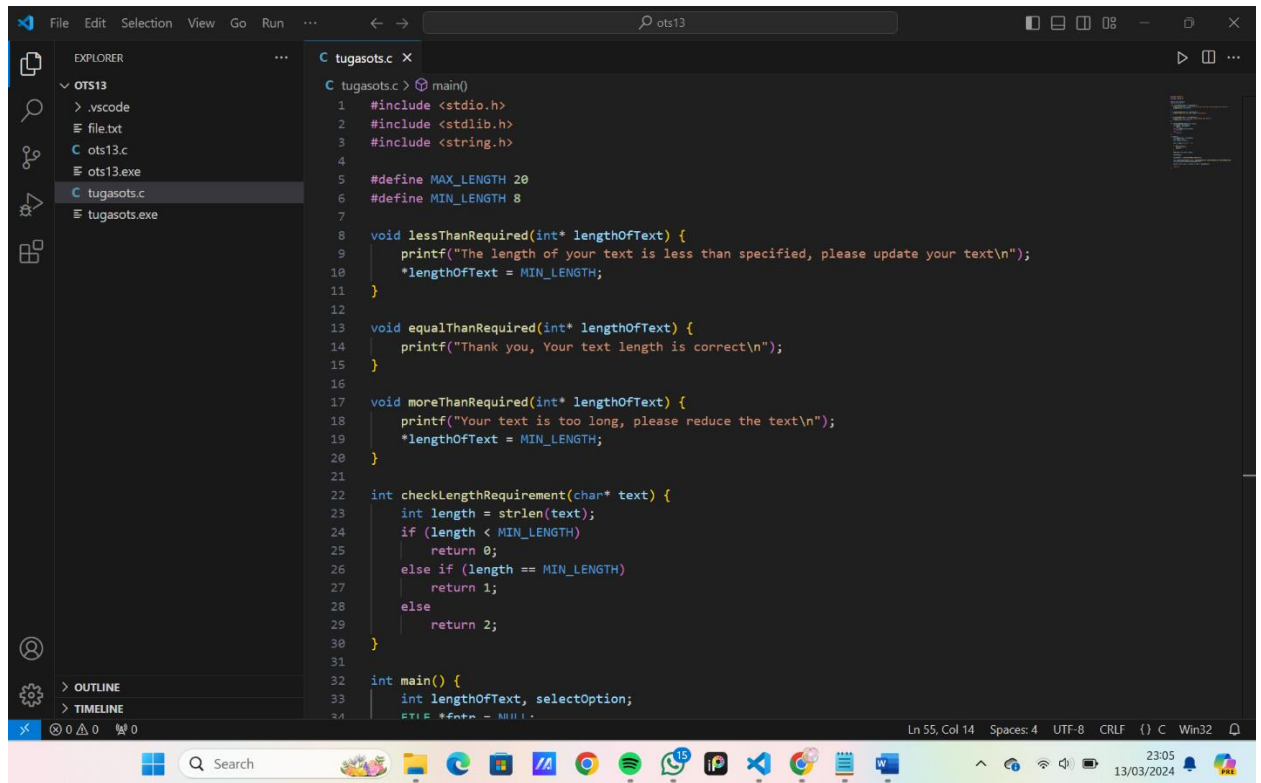
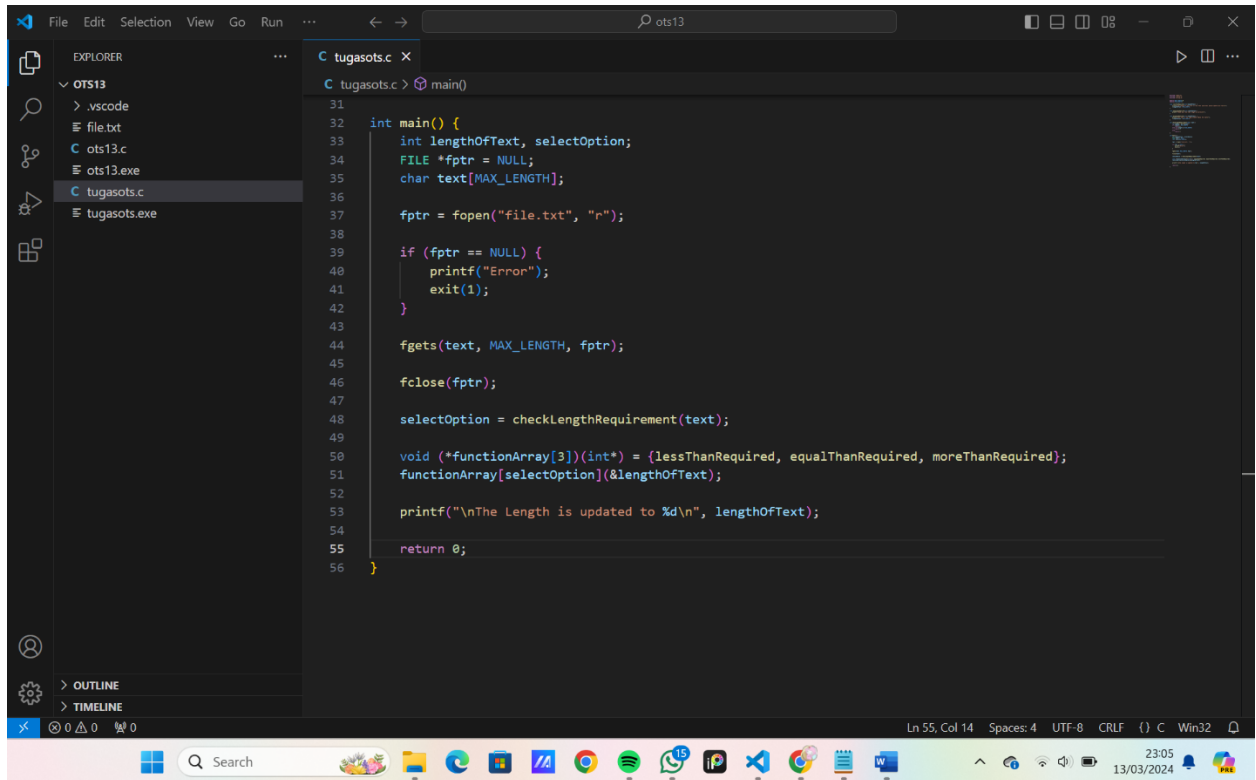


Kelas : IF 03-03





## 2. Penjelasan

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

Mendeklarasikan library standar untuk proses input-output untuk fungsi printf dan scanf

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

Mengimpor library standar untuk fungsi-fungsi umum seperti exit

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

Mendeklarasikan library untuk fungsi manipulasi string

```
#define MAX_LENGTH 20
```

Mendefinisikan panjang maksimum karakter adalah 20

```
#define MIN_LENGTH 8
```

Mendefinisikan panjang minimum karakter adalah 8

```
void lessThanRequired(int* lengthOfText) {
```

Deklarasi sebuah fungsi bernama 'lessThanRequired' yang mengambil satu parameter, yaitu pointer ke variabel bertipe integer yang disebut 'lengthOfText'. Fungsi ini tidak mengembalikan nilai (void). Saat fungsi 'lessThanRequired' dipanggil, kita harus memberikan alamat memori dari

sebuah variabel integer. Isi variabel yang ditunjuk oleh pointer tersebut dapat diubah di dalam fungsi

```
printf("The length of your text is less than specified, please update your text\n");
```

Mencetak pesan ke layar dan memberi tahu pengguna bahwa panjang teks yang dimasukkan terlalu pendek

```
*lengthOfText = MIN_LENGTH;
```

Mengubah nilai variabel yang ditunjuk oleh pointer 'lengthOfText' dengan nilai dari konstanta 'MIN\_LENGTH'. Operator dereferensi '\*' membantu mengakses nilai yang disimpan di alamat memori yang ditunjuk oleh pointer. Kemudian nilai tersebut diganti dengan nilai dari konstanta 'MIN\_LENGTH'. Hasilnya, variabel yang ditunjuk oleh 'lengthOfText' akan memiliki nilai yang sama dengan 'MIN\_LENGTH'. Ini berguna untuk mengatur ulang panjang teks menjadi nilai minimum yang diinginkan dalam situasi dimana panjang teks terlalu pendek atau terlalu panjang

```
void equalThanRequired(int* lengthOfText) {
```

Fungsi ini digunakan dalam situasi dimana panjang teks sama dengan panjang yang diinginkan ('MIN\_LENGTH'). Dalam implementasinya, fungsi ini hanya mencetak pesan ke layar yang menyatakan bahwa panjang teks sudah sesuai dengan yang diinginkan. Diindikasikan oleh parameter 'lengthOfText' yang merupakan pointer ke variabel bertipe integer, tetapi nilai variabel tersebut tidak diubah dalam fungsi ini

```
printf("Thank you, Your text length is correct\n");
```

Mencetak pesan dan memberi tahu pengguna bahwa panjang teks sudah sesuai

```
void moreThanRequired(int* lengthOfText) {
```

Fungsi ini digunakan untuk menangani situasi dimana panjang teks melebihi panjang maksimum yang diinginkan ('MAX\_LENGTH')

```
printf("Your text is too long, please reduce the text\n");
```

Mencetak pesan dan memberi tahu pengguna bahwa teks terlalu panjang

```
*lengthOfText = MIN_LENGTH;
```

Mengubah nilai variabel yang ditunjuk oleh pointer 'lengthOfText' menjadi 'MIN\_LENGTH'. Nilai yang disimpan di alamat memori yang ditunjuk oleh pointer dapat diakses dengan menggunakan operator dereferensi '\*', kemudian mengganti nilainya dengan nilai dari konstanta 'MIN\_LENGTH'

```
int checkLengthRequirement(char* text) {
```

Fungsi ini mengembalikan nilai integer, menerima sebuah pointer ke karakter ('char\*') yang merupakan teks, dan memeriksa panjang teks tersebut

```
int length = strlen(text);
```

Menghitung panjang karakter dari string yang disimpan dalam variabel 'text' dan hasilnya disimpan dalam variabel 'length'

```
if (length < MIN_LENGTH)
```

Mengecek apakah panjang string atau nilai variabel 'length' kurang dari nilai minimum yang ditentukan 'MIN\_LENGTH'

```
return 0;
```

Mengembalikan nilai 0 jika panjang teks kurang dari 'MIN\_LENGTH'

```
else if (length == MIN_LENGTH)
```

Mengecek apakah panjang string atau nilai variabel 'length' sama dengan nilai minimum yang ditentukan 'MIN\_LENGTH'

```
return 1;
```

Mengembalikan nilai 1 jika panjangnya sama dengan 'MIN\_LENGTH'

```
else
```

Mengecek jika panjang teks lebih dari 'MIN\_LENGTH'

```
return 2;
```

Mengembalikan nilai 2 jika panjangnya lebih besar dari 'MIN\_LENGTH'

```
int main() {
```

Deklarasi fungsi utama program

```
int lengthOfText, selectOption;
```

Deklarasi dua variabel bertipe integer 'lengthOfText' dan 'selectOption'. Dua variabel ini digunakan untuk menyimpan informasi terkait panjang teks dan opsi yang dipilih dalam program

```
FILE *fptr = NULL;
```

Deklarasi pointer 'fptr' ke tipe data 'FILE' yang diinisialisasi dengan nilai 'NULL'. Pointer ini digunakan untuk mengakses file dalam program

```
char text[MAX_LENGTH];
```

Mendeklarasikan array 'text' yang bertipe 'char' dengan panjang 'MAX\_LENGTH'. Array ini digunakan untuk menyimpan teks yang akan diproses dalam program

```
fptr = fopen("file.txt", "r");
```

Membuka file bernama "file.txt" dalam mode baca "r" dan menugaskan file tersebut ke pointer 'fptr' sehingga 'fptr' menunjuk ke file yang terbuka

```
if (fptr == NULL) {
```

Memeriksa apakah pointer 'fptr' memiliki nilai 'NULL' yang menunjukkan bahwa pembukaan file tidak berhasil

```
printf("Error");
```

Mencetak pesan "Error" ke layar dan memberi tahu pengguna bahwa pembukaan file gagal, menandakan bahwa ada kesalahan dalam proses membuka file

```
exit(1);
```

Mengakhiri program dengan keluaran 1 yang menandakan bahwa program mengalami kesalahan

```
fgets(text, MAX_LENGTH, fptr);
```

Membaca baris teks dari file yang ditunjuk pointer 'fptr' dengan panjang maksimum 'MAX\_LENGTH' dan menyimpannya ke array 'text'

```
fclose(fptr);
```

Menutup file yang ditunjuk oleh pointer 'fptr' setelah selesai membacanya

```
selectOption = checkLengthRequirement(text);
```

Memanggil fungsi 'checkLengthRequirement' untuk memeriksa panjang teks yang disimpan dalam variabel 'text'. Hasilnya akan disimpan dalam variabel 'selectOption' yang akan digunakan untuk menentukan apa yang harus dilakukan selanjutnya dalam program

```
void (*functionArray[3])(int*) = {lessThanRequired, equalThanRequired, moreThanRequired};
```

Mendeklarasikan array yang berisi tiga pointer ke fungsi-fungsi yang masing-masing menerima satu parameter, yaitu pointer ke integer. Array ini digunakan untuk menyimpan tiga fungsi berbeda yaitu 'lessThanRequired', 'equalThanRequired', dan 'moreThanRequired'

```
functionArray[selectOption] (&lengthOfText) ;
```

Memanggil salah satu fungsi dari array 'functionArray' berdasarkan nilai variabel 'selectOption'. Fungsi yang dipanggil akan menerima satu parameter, yaitu alamat memori dari variabel 'lengthOfText', dan kemudian akan dieksekusi sesuai dengan kondisi yang ada berdasarkan nilai 'selectOption'

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

Mencetak pesan ke layar dan memberi tahu pengguna bahwa panjang teks telah diperbarui. Nilai panjang teks 'lengthOfText' akan dimasukkan ke dalam pesan menggunakan format '%d' sehingga nilai panjang teks akan ditampilkan di dalam pesan

```
return 0;
```

Mengembalikan nilai 0 dari fungsi 'main()' dan menandakan bahwa program telah berjalan dengan sukses tanpa kesalahan

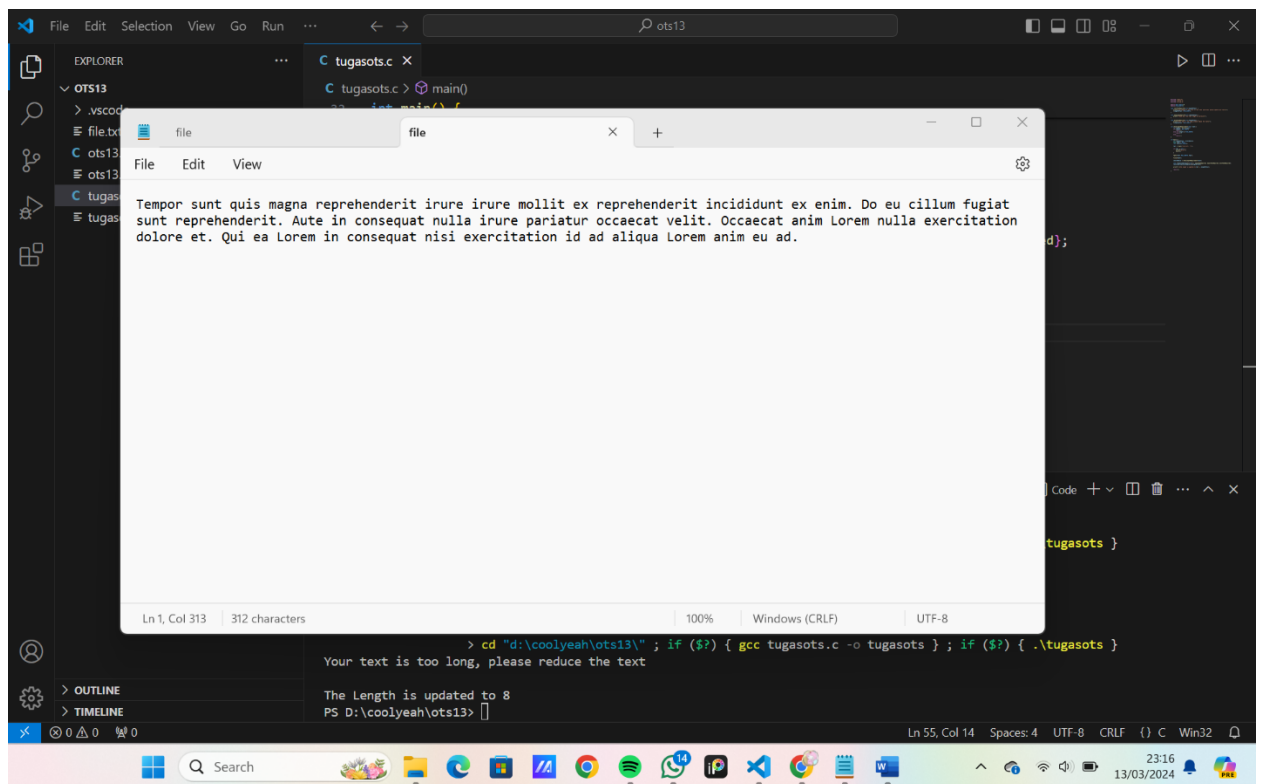
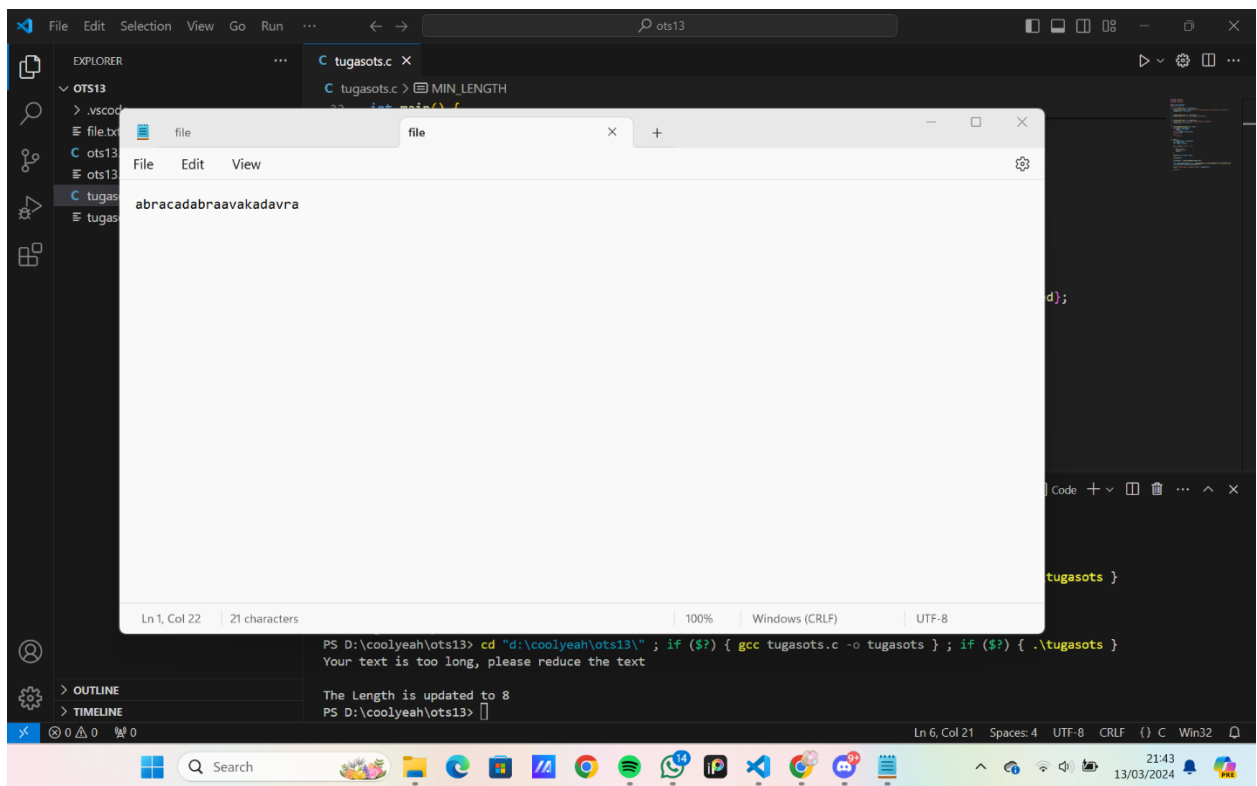
### 3. Output

The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying the project structure. The main editor window shows the source code of `tugasots.c`. A terminal window is open in the foreground, displaying the output of the program. The input text is `mariaayu`, and the output message is `Thank you, Your text length is correct`. The terminal also shows the command `cd "d:\coolyeah\ots13\" ; gcc tugasots.c -o tugasots ; if ($?) { .\tugasots }` and the result `The Length is updated to 16977280`.

```
PS D:\coolyeah\ots13> cd "d:\coolyeah\ots13\" ; gcc tugasots.c -o tugasots ; if ($?) { .\tugasots }
Thank you, Your text length is correct
The Length is updated to 16977280
PS D:\coolyeah\ots13>
```

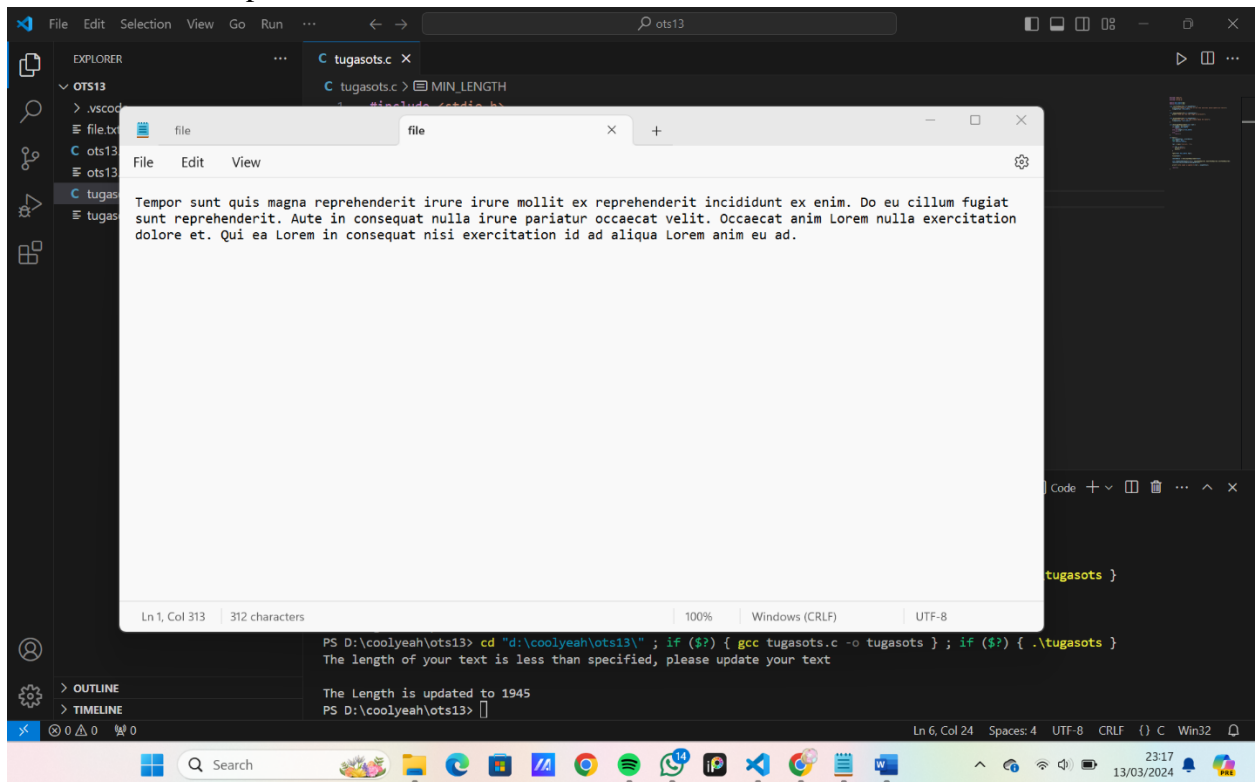
The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying the project structure. The main editor window shows the source code of `tugasots.c`. A terminal window is open in the foreground, displaying the output of the program. The input text is `bali`, and the output message is `The Length of your text is less than specified, please update your text`. The terminal also shows the command `cd "d:\coolyeah\ots13\" ; gcc tugasots.c -o tugasots ; if ($?) { .\tugasots }` and the result `The Length is updated to 8`.

```
PS D:\coolyeah\ots13> cd "d:\coolyeah\ots13\" ; gcc tugasots.c -o tugasots ; if ($?) { .\tugasots }
The Length of your text is less than specified, please update your text
The Length is updated to 8
PS D:\coolyeah\ots13>
```





Berikut ini adalah output ketika MAX\_LENGTH 2024 dan MIN\_LENGTH 1945



The screenshot shows a Visual Studio Code editor window with a C program named `tugasots.c`. The program is a simple C program that prints a fixed string of Lorem Ipsum text. The string is: `Tempor sunt quis magna reprehenderit irure irure mollit ex reprehenderit incididunt ex enim. Do eu cillum fugiat sunt reprehenderit. Aute in consequat nulla irure pariatur occaecat velit. Occaecat anim Lorem nulla exercitation dolore et. Qui ea Lorem in consequat nisi exercitation id ad aliqua Lorem anim eu ad.`

The output of the program is displayed in the terminal window at the bottom of the editor. The output shows the command `cd "d:\coolyeah\ots13"` and the command `gcc tugasots.c -o tugasots` being executed. The output also shows the command `if ($?) { gcc tugasots.c -o tugasots } ; if ($?) { .\tugasots }` being executed. The output of the program is: `The length of your text is less than specified, please update your text`. The output also shows the command `The Length is updated to 1945` being executed.

The status bar at the bottom of the editor shows the current line and column as `Ln 6, Col 24`, the number of spaces as `Spaces: 4`, the encoding as `UTF-8`, the line ending as `CRLF`, and the file name as `( ) C Win32`. The system tray at the bottom right shows the time as `23:17` and the date as `13/03/2024`.