

**1 D4 - TEKKOM B**



**LAPORAN RESMI**

**VARIABEL DAN FUNGSI STRING**



Nama : Septian Bagus Jumantoro

Kelas : 1 – D4 Teknik Komputer B

NRP : 3221600039

Dosen : Ir Sigit Wasista M.Kom.

Mata Kuliah : Praktikum Pemrograman Dasar 1

Hari/Tgl. Praktikum : 22 Oktober 2021

PERCOBAAN 8

* 1. JUDUL: VARIABEL & FUNGSI STRING
  2. TUJUAN: Mempelajari penggunaan variable string yang dipadukan dengan fungsi-fungsi untuk pengolahan data string, seperti strcat(), strlen(), strcpy dan sebagainya.
  3. TEORI: Suatu karakter string adalah terdiri dari beberapa karakter (deretan karakter) yang diakhiri dengan karakter khusus ‘\0’ (null). Suatu konstanta string adalah karakter string yang dituliskan diantara tanda petik ganda (“), seperti yang digunakan pada fungsi printf() atau puts(). Konstanta karakter string mempunyai kelas penyimpan static, sehingga bila digunakan didalam fungsi, maka nilainya tidak akan berubah selama program aktif.
  4. PROGRAM PERCOBAAN
     1. Mengamati penggunaan variable array berukuran bebas “[]” untuk melewatkan suatu variable string ke dalam fungsi. Untuk dapat melewatkan suatu nilai konstanta atau variable yang tidak diketahui panjangnya, maka digunakan array dengan ukuran bebas. Meskipun digunakan variable berukuran kosong, compiler akan tahu berapa jumlah elemen yang dimasukkan dalam suatu variable, seperti contoh dibawah. Program dibawah ini adalah untuk menghitung berapa panjang dari karakter string, dengan acuan karakter ‘\0’ (null).

/\* Nama File : STRING1.C \*/

int panjang\_str(c!ar string[])

{

int !itung=O;

w!ile (string[!itung] != '\O') ++!itung; return (!itung);

}

void main()

{

static c!ar kataO[]={'b','e','l','a','j','a','r','\O'};

static c!ar kata1[]={'f','u','n','g','s','i','\O'};

static c!ar kata2[]={'s','t','r','i','n','g','\O'};

printf("Panjang string O = %d\n",panjang\_str(kataO)); printf("Panjang string 1 = %d\n",panjang\_str(kata1)); printf("Panjang string 2 = %d\n",panjang\_str(kata2));

getc!();

}

* + 1. Mengamati penggunaan fungsi-fungsi untuk pengolahan string, seperti strlen(), strcat() dan strcmp(). Fungsi strlen() digunakan untuk menghitung jumlah string seperti pada program 8.4.1 diatas, fungsi strcat() digunakan untuk menggabungkan antara dua string, fungsi strcmp() digunakan untuk membandingkan antara dua string, diberikan nilai 0 bila kedua string tidak sama. Penggunaan variable static mutlak diperlukan apabila menggunakan variable string dengan ukuran bebas, bila tidak digunakan variable static maka kemungkinan terjadi kesalahan isi dari variable dengan ukuran bebas tersebut, dimana karakter yang diberikan tidak dapat masuk dengan sempurna, atau diisi dengan karakter sembarang.

/\* Nama File : STRING2.C \*/

#include cstring.!S #include cstdio.!S

void main()

{

c!ar nama[5O], jawab[15];

static c!ar salam[]=", Assalamualaikum..."; static c!ar wayang[]="BimaSena";

puts("Masukkan nama anda:"); gets(nama);

puts("");

puts("Siapaka! nama wayang yang gak bisa duduk?"); gets(jawab);

puts(""); if(strlen(salam)+strlen(nama)c=5O)

strcat(nama,salam);

puts(nama); if(strcmp(wayang,jawab)==O)

printf("\nJawaban anda benar! Hebat!"); else {

printf("\nJawaban anda sala!!"); printf("\nJawabannya adala! %s",wayang);

}

puts("");

getc!();

}

* + 1. Menghitung jumlah karakter yang dimasukkan melalui keyboard, menggunakan fungsi fgets(). Fungsi ini dapat juga digunakan untuk membaca data dari file, dengan mengganti standar masukan yang disini ditulis stdin, dimana masukan didapakkan dari keyboard.

/\* Nama File : STRING3.C \*/ #include cstdio.!S

#define MAKS 256

void main()

{

int i, jumkar; c!ar teks[MAKS];

puts("Masukkan suatu kalimat (maks 255 karakter)."); puts("Komputer akan meng!itung jumla! karakternya.\n"); fgets(teks, sizeof teks, stdin);

jumkar = O;

for(i=O; teks[i]; i++) jumkar++;

printf("\nJumla! karakter = %d\n", jumkar); getc!();

}

* + 1. Mencari posisi karakter pada suatu string menggunakan fungsi

strc!r().

/\* Nama File : STRING4.C \*/ #include cstring.!S

void main()

{

c!ar str1[5O], str2[5O], kar, \*po;

strcpy(str1,"O 1 2 3"); strcpy(str2,"-S KUCARI KAMU UNTUK KUTANGKAP.");

cputs("Isi dari str1 = "); puts(str1); cputs("Isi dari str2 = "); puts(str2);

printf("\nPosisi !uruf pada string dapat dicari"); printf("\nmenggunakan rumus p = strc!r(str2,kar)\n"); kar='G';

po=strc!r(str2,kar)-str2;

printf("\nKarakter '%c' terletak pada kolom ke %d",kar,po);

getc!();

}

* + 1. Karakter string adalah semua karakter yang digunakan oleh komputer, termasuk angka, tanda baca, tanda khusus dan lain-lain. Program dibawah mencoba untuk memisahkan karakter huruf (alphabet) dengan karakter lainnya, sehingga dapat dihitung jumlah huruf yang terdapat dalam suatu string. Fungsi alp!abetic() akan memberikan nilai balik 1 bila ditemukan karakter huruf dalam suatu string. Fungsi !itung\_kata() akan memanfaatkan fungsi alp!abetic() untuk menghitung kata yang ada dalam suatu string.

/\* Nama File : STRING5.C \*/

int alp!abetic(c!ar c)

{

if ((cS='a' && cc='z') || (cS='A' && cc='Z')) return(1);

else

return(O);

}

int !itung\_kata(c!ar string[])

{

int i, li!at\_kata=1, kata\_ter!itung=O;

for (i=O; string[i]!='\O'; ++i)

if (alp!abetic(string[i])) { if (li!at\_kata) {

++kata\_ter!itung; li!at\_kata=O;

}

} else

li!at\_kata=1;

return (kata\_ter!itung);

}

void main()

{

int !itung;

static c!ar teks1[] = "Angka atau bilangan O123456789"; static c!ar teks2[] = "juga termasuk karakter string.";

printf("Teks 1: %s\n",teks1);

printf("Teks 2: %s\n",teks2); puts("");

!itung = !itung\_kata(teks1);

!itung = !itung + !itung\_kata(teks2);

printf("Ter!itung ada %d kata pada teks diatas!\n",!itung);

getc!();

}

* + 1. Mengamati pengolahan string menggunakan fungsi strncmp(), dimana digunakan untuk mengambil bagian kata dari suatu kalimat dalam string. Untuk menggambil kata pada string pada kolom tertentu dengan panjang tertentu, maka dapat dilakukan dengan menambah suatu konstanta yang menunjukkan posisi kolom yang dimaksud, misalkan pada kata “dulu” dalam string “Don’t dulu bekerja”, maka dapat dilakukan dengan menambah suatu konstanta dimana posisi kolom kata “dulu” berada, yaitu dengan menghitung dari kiri D=0, o=1, n=2 dst, sehingga posisi d pada kata “dulu” adalah 6, sehingga posisinya adalah str1+6.

/\* Nama File : STRING6.C \*/ #includecstdlib.!S

void main()

{

c!ar str1[3O], str2[3O], str3[3O];

c!ar str4[3O], str5[3O], str6[3O];

strcpy(str1,"Don't dulu bekerja"); strcpy(str2,"baru sambil be !appy"); strcpy(str3,"Ini worry Belajar itu");

cputs("str1 = "); puts(str1);

cputs("str2 = "); puts(str2);

cputs("str3 = "); puts(str3);

cputs("\nTekan ENTER untuk meli!at !asilnya!"); getc!();

strncpy(str4,str3,4); str4[4] = NULL;

strncpy(str5,str1+6,5); str5[5] = NULL;

strncpy(str6,str2,5); str6[5] = NULL;

strcat(str4,str5); strcat(str4,str6); strncpy(str6,str3+18,3); str6[3] = NULL;

strcat(str4,str6);

printf("\n\nIsi dari str4 = "); puts(str4);

getc!();

}

* + 1. Fungsi yang digunakan untuk membalik suatu kata/ kalimat adalah fungsi strrev(), suatu kata/ kalimat dikatakan palindrome apabila dibaca dari depan maupun belakang sama, misalkan KAKAK, KAPAK, KATAK, dan sebagainya .

/\* Nama File : STRING7.C \*/

#include cstring.!S #include cstdio.!S #include cconio.!S

void main()

{

c!ar kalimat[5O];

printf("Masukkan kata/ kalimat : "); scanf("%s",kalimat);

puts("");

printf("\nKata/ kalimat sebelum strrev():\n"); printf("%s\n\n",kalimat);

printf("\nKata/ kalimat setela! strrev():\n"); strrev(kalimat);

printf("%s\n",kalimat);

getc!();

}

* 1. TUGAS-TUGAS (Dikumpulkan 1 minggu setelah praktikum dilaksanakan)
     1. Buatlah fungsi yang disebut strsisip() untuk menyisipkan satu karakter string kedalam string lainnya. Argumen untuk fungsi harus berisi string sumber, string yang disisipkan, dan posisi dari string sumber dimana string disisipkan. Jadi dapat dipanggil seperti dibawah ini:

strsisip(teks,"se",6);

dengan teks adalah kalimat "sala! orang".

Jadi setelah penyisipan kata menjadi "sala! seorang".

* **Source Code**

//tugas 8.1

#include <stdio.h>

#include <string.h>

char a[100], kata[100],b[100];

char strsisip(char kata[],char a[],int z){

strncpy(b,kata,z);

strncat(b+z,a,strlen(a)-1);

strcat(b+z+strlen(a)-1,kata+z);

return (0);

}

int main()

{

    system("cls");

    int z;

    printf("Masukkan teks: ");

    fgets(kata,sizeof kata,stdin);

    puts(" ");

    printf("kata yang akan disisipkan: ");

    fgets(a,sizeof a,stdin);

    puts(" ");

    printf("Disisipkan setelah karakter ke: ");

    scanf("%d",&z);

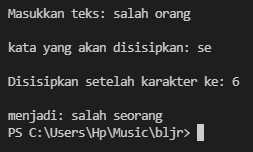
    strsisip(kata,a,z);

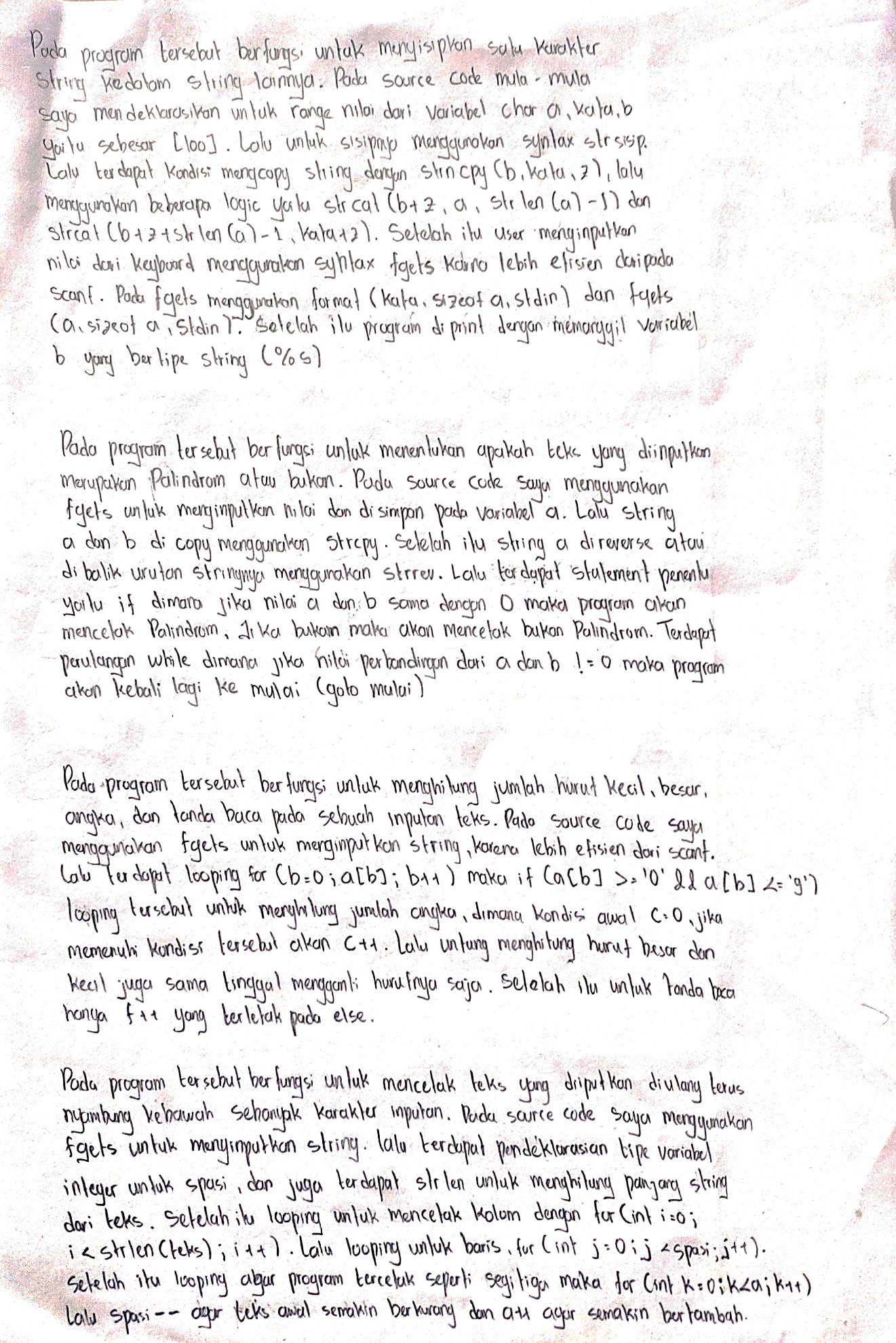
    puts(" ");

    printf("menjadi: %s", b);

}

* **Output**



* **Analisa**
  + 1. Buatlah program untuk menentukan suatu kata/ kalimat merupakan suatu polindrom atau bukan. Dimana kata/ kalimat tersebut dimasukkan melalui keyboard.

Contoh tampilan :

Kalimat : KASUR RUSAK Termasuk PALINDROM

Kalimat : MAKAN MALAM Bukan PALINDROM

* **Source Code**

//tugas 8.2

#include <stdio.h>

#include <string.h>

void main()

{

    system("cls");

    char a[20], b[20];

    mulai:

    printf("Masukkan teks: ");

    gets (a);

    strcpy(b,a);

    strrev(a);

    if (strcmp(a,b) == 0)

    {

        printf("\nMerupakan Palindrom");

    }

    else

    {

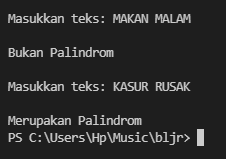
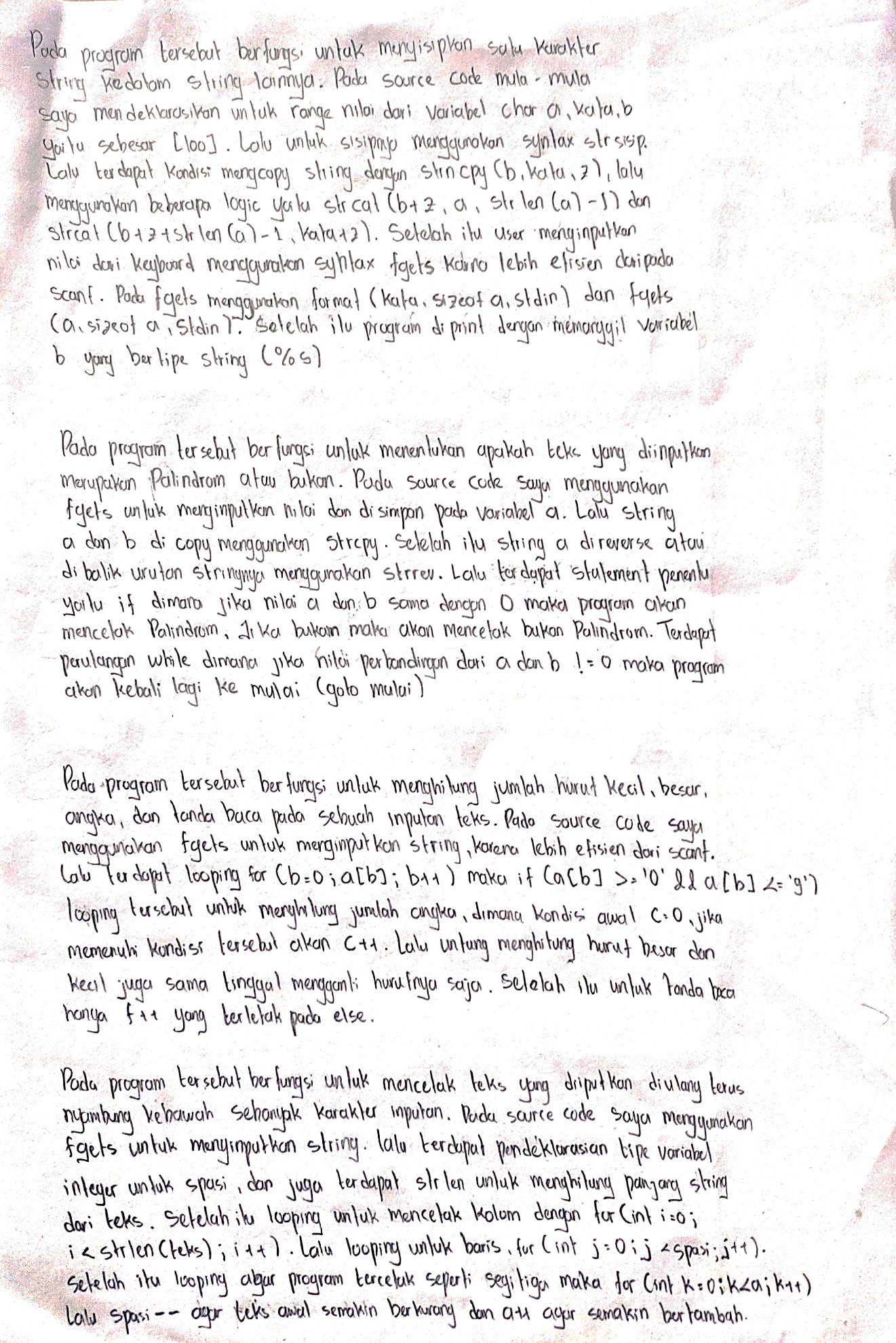
        printf("\nBukan Palindrom\n");

    }

    while (strcmp(a,b) != 0)

    goto mulai;

}

* **Output**
* **Analisa** 
  + 1. Buatlah program untuk menghitung jumlah huruf kecil, huruf besar, angka numerik, dan tanda baca didalam suatu string yang dimasukkan melalui keyboard.
* **Source Code**

//tugas 8.3

#include <stdio.h>

#include <string.h>

void main(){

    system("cls");

    char a[100];

    int b, c=0, d=0, e=0, f=-1;

    printf("Masukkan sebuah teks: ");

    fgets(a,sizeof a,stdin);

    for(b=0; a[b]; b++){

        if(a[b] >= '0' && a[b] <= '9')

            c++;

        else{

                if(a[b] >= 'a' && a[b] <= 'z')

                    d++;

                else{

                        if(a[b] >= 'A' && a[b] <= 'Z')

                            e++;

                        else

                            f++;

                    }

            }

    }

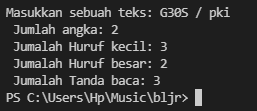
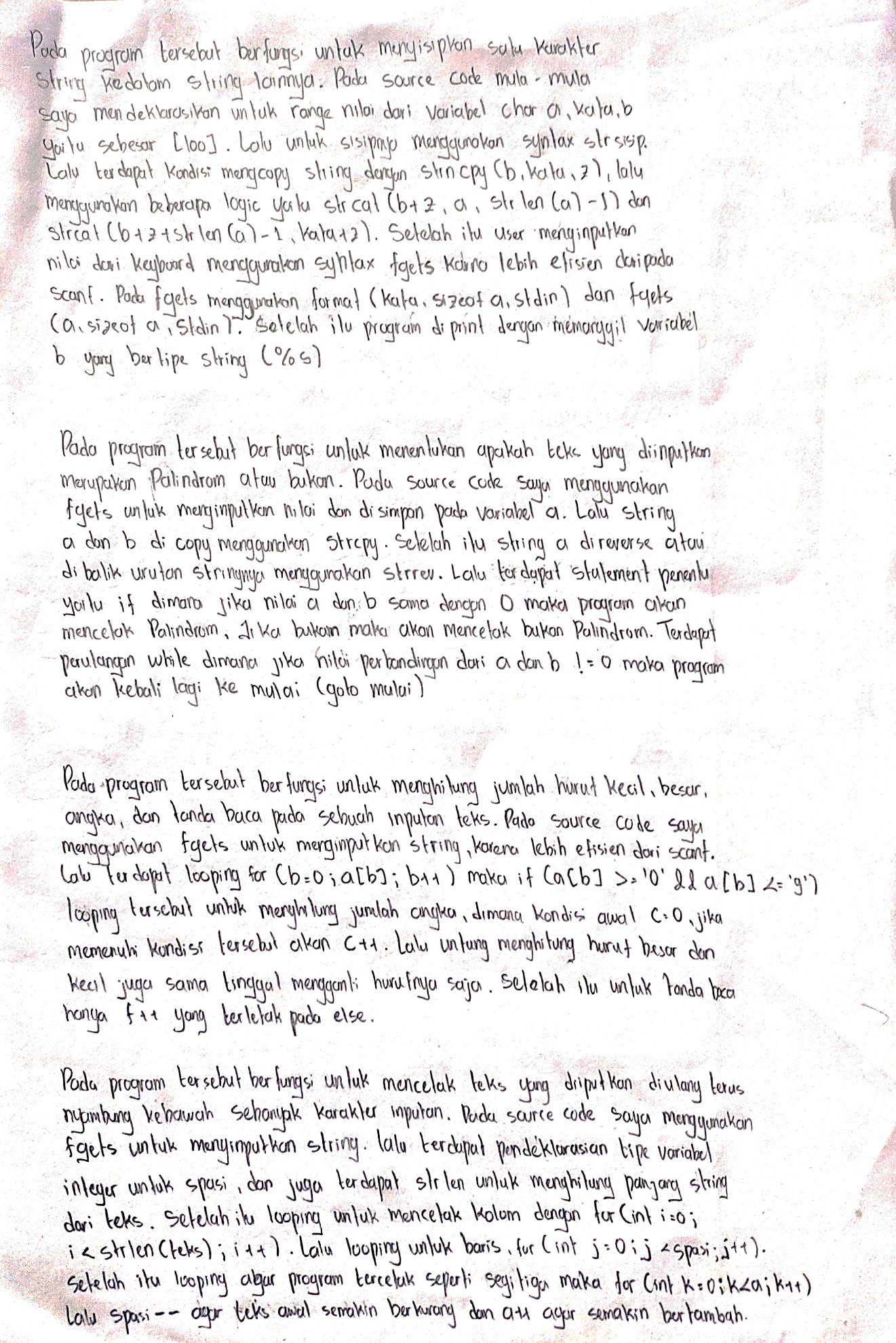
    printf(" Jumlah angka: %d\n",c);

    printf(" Jumalah Huruf kecil: %d\n",d);

    printf(" Jumalah Huruf besar: %d\n",e);

    printf(" Jumalah Tanda baca: %d\n",f);

}

* **Output**
* **Analisa** 
  + 1. Buatlah program untuk menghasilkan keluaran sebagai berikut :

p PO POL POLI POLIT

POLITE POLITEK POLITEKN POLITEKNI POLITEKNIK

POLITEKNIK POLITEKNIK E

POLITEKNIK EL POLITEKNIK ELE POLITEKNIK ELEK POLITEKNIK ELEKT POLITEKNIK ELEKTR POLITEKNIK ELEKTRO POLITEKNIK ELEKTRON POLITEKNIK ELEKTRONI POLITEKNIK ELEKTRONIK POLITEKNIK ELEKTRONIKA

POLITEKNIK ELEKTRONIKA POLITEKNIK ELEKTRONIKA N

POLITEKNIK ELEKTRONIKA NE POLITEKNIK ELEKTRONIKA NEG POLITEKNIK ELEKTRONIKA NEGE POLITEKNIK ELEKTRONIKA NEGER POLITEKNIK ELEKTRONIKA NEGERI

POLITEKNIK ELEKTRONIKA NEGERI POLITEKNIK ELEKTRONIKA NEGERI S

POLITEKNIK ELEKTRONIKA NEGERI SU POLITEKNIK ELEKTRONIKA NEGERI SUR POLITEKNIK ELEKTRONIKA NEGERI SURA POLITEKNIK ELEKTRONIKA NEGERI SURAB POLITEKNIK ELEKTRONIKA NEGERI SURABA POLITEKNIK ELEKTRONIKA NEGERI SURABAY POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

Setelah tampilan diatas didapatkan, maka modifikasilah program dengan menampilkan tiap baris tersebut pada posisi koordinat tertentu misalkan koordinat (56,10). Dimana baris pertama menempati koordinat awal, kemudian baris kedua menempati koordinat (55,10), kemudian baris ketiga (54,10), keempat (53,10) dan seterusnya sampai baris terakhir menempati koordinat (24,10). Gunakan fungsi delay(1000) untuk engatur tampilan baris satu dengan lainnya.

* **Source Code**

#include <stdio.h>

#include <string.h>

void main()

{

    static char str1[] = "POLITEKNIK ELEKTRONIKA NEGERI SURABAYA";

    int d,a;

    system("cls");

    d = strlen(str1);

    a = d;

    for(int i=0; i<=d; i++)

    {

        for(int b=a; b>0; b--)

        {

            printf(" ");

        }

        a--;

        for(int j=0; j<i; j++)

        {

            printf("%c",str1[j]);

        }

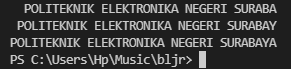
    puts("");

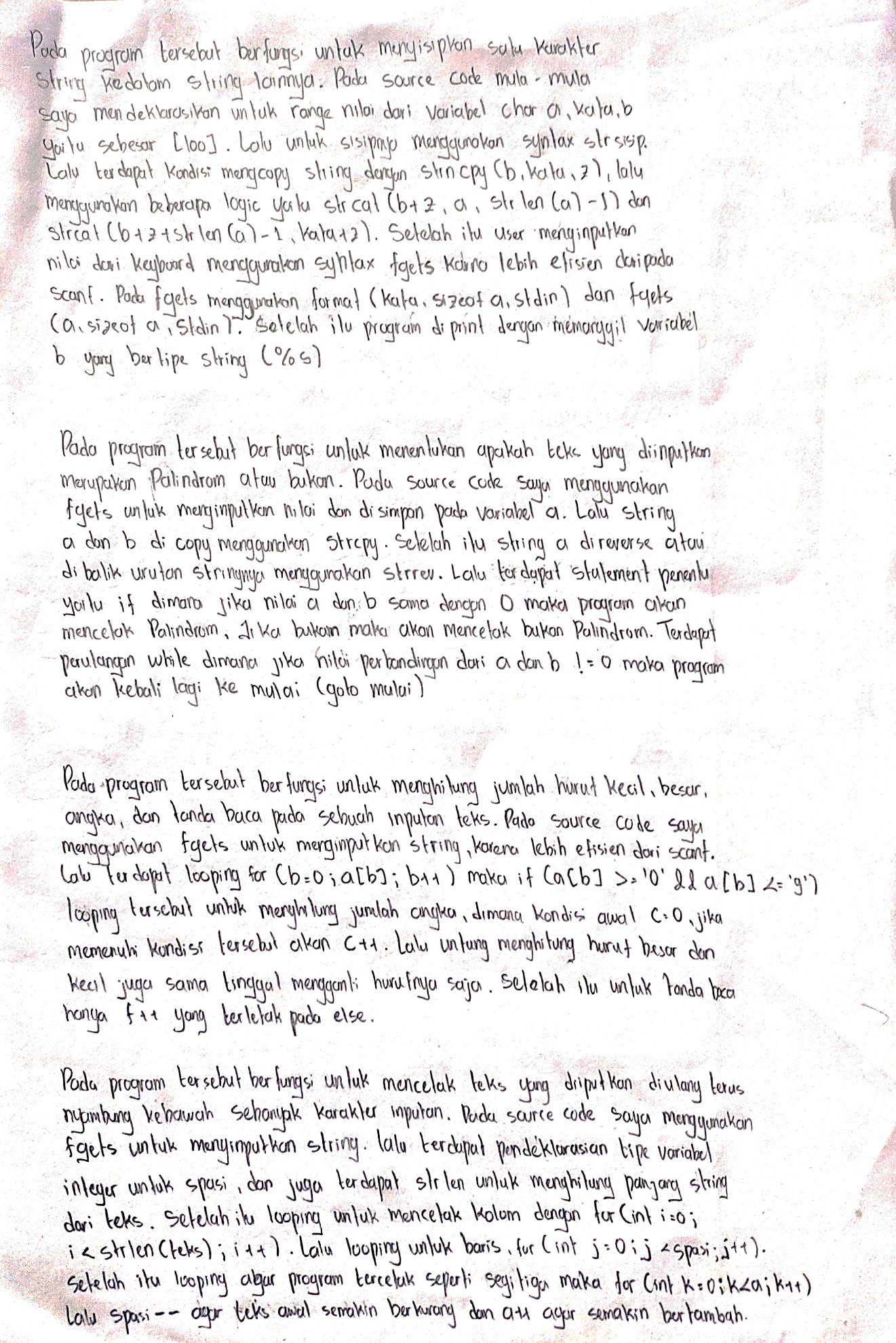
    }

}

* **Output**





* **Analisa**