

1 D4 - TEKKOM B

LAPORAN RESMI VARIABEL DAN FUNGSI STRING



Nama	:	Septian Bagus Jumanoro
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

8.1 JUDUL: VARIABEL & FUNGSI STRING

8.2 TUJUAN: Mempelajari penggunaan variable string yang dipadukan dengan fungsi-fungsi untuk pengolahan data string, seperti `strcat()`, `strlen()`, `strcpy` dan sebagainya.

8.3 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 penyimpanan static, sehingga bila digunakan didalam fungsi, maka nilainya tidak akan berubah selama program aktif.

8.4 PROGRAM PERCOBAAN

8.4.1 Mengamati penggunaan variable array berukuran bebas `"[]"` untuk melewati suatu variable string ke dalam fungsi. Untuk dapat melewati 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(char string[])
{
    int !itung=0;

    while (string[!itung] != '\0') ++!itung;
    return (!itung);
}

void main()
{
    static char kata0[]={'b','e','l','a','j','a','r','\0'};
```

8.4.2 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[50], 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)<50)

        strcat(nama,salam);

    puts(nama):
```

8.4.3 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 didapatkan dari keyboard.

```
/* Nama File : STRING3.C */

#include cstdio.h

#define MAKS 256

void main()
{
    int i, jumkar;
    char teks[MAKS];

    puts("Masukkan suatu kalimat (maks 255 karakter).");
    puts("Komputer akan menghitung jumlah karakternya.\n");
    fgets(teks, sizeof teks, stdin);

    jumkar = 0;
```

8.4.4 Mencari posisi karakter pada suatu string menggunakan fungsi `strcbr()`.

```
/* Nama File : STRING4.C */

#include cstring.h

void main()
{
    char str1[50], str2[50], kar, *po;

    strcpy(str1, "0.....1.....2.....3");
    strcpy(str2, "-S KUCARI KAMU UNTUK KUTANGKAP.");

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

    printf("\nPosisi huruf pada string dapat dicari");
    printf("\nmenggunakan rumus p = strcbr(str2, kar)\n");
```

8.4.5 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(0);
}

int !itung_kata(c!ar string[])
{
    int i, li!at_kata=1, kata_ter!itung=0;

    for (i=0; string[i]!='\0'; ++i)
        if (alp!abetic(string[i])) {
            if (li!at_kata) {
                ++kata_ter!itung;
                li!at_kata=0;
            }
        } else
            li!at_kata=1;

    return (kata_ter!itung);
}
```

8.4.6 Mengamati pengolahan string menggunakan fungsi `strncpy()`, dimana digunakan untuk mengambil bagian kata dari suatu kalimat dalam string. Untuk mengambil 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 */

#include <stdlib.h>

void main()
{
    char str1[30], str2[30], str3[30];
    char str4[30], str5[30], str6[30];

    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!");
    getch();

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

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

8.4.7 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 <string.h>
#include <stdio.h>
#include <conio.h>

void main()
{
    char kalimat[50];

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

    puts("");

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

```

8.5 TUGAS-TUGAS (Dikumpulkan 1 minggu setelah praktikum dilaksanakan)

8.5.1 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);
    strcat(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

```

Masukkan teks: salah orang

kata yang akan disisipkan: se

Disisipkan setelah karakter ke: 6

menjadi: salah seorang
PS C:\Users\Hp\Music\bljr>

```

➤ Analisa

Pada program tersebut berfungsi untuk menyisipkan satu karakter string kedalam string lainnya. Pada source code mula-mula saya mendeklarasikan untuk range nilai dari variabel char a, kata, b yaitu sebesar [100]. Lalu untuk sisipnya menggunakan syntax strsisip. Lalu terdapat kondisi mengcopy string dengan strcpy(b,kata,z), lalu menggunakan beberapa logika yaitu strcat(b+z,a,strlen(a)-1) dan strcat(b+z+strlen(a)-1,kata+z). Setelah itu user memasukkan nilai dari keyboard menggunakan syntax fgets karena lebih efisien daripada scanf. Pada fgets menggunakan format (kata,sizeof a,stdin) dan fgets(a,sizeof a,stdin). Setelah itu program di print dengan memanggil variabel b yang ber tipe string (%s)

8.5.2 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

```
Masukkan teks: MAKAN MALAM

Bukan Palindrom

Masukkan teks: KASUR RUSAK

Merupakan Palindrom
PS C:\Users\Hp\Music\bljr> █
```



```

    }
}

printf(" Jumlah angka: %d\n",c);
printf(" Jumlah Huruf kecil: %d\n",d);
printf(" Jumlah Huruf besar: %d\n",e);
printf(" Jumlah Tanda baca: %d\n",f);
}

```

➤ Output

```

Masukkan sebuah teks: G30S / pki
Jumlah angka: 2
Jumlah Huruf kecil: 3
Jumlah Huruf besar: 2
Jumlah Tanda baca: 3
PS C:\Users\Hp\Music\bljr>

```

➤ Analisa

Pada program tersebut berfungsi untuk menghitung jumlah huruf kecil, besar, angka, dan tanda baca pada sebuah inputan teks. Pada source code saya menggunakan fgets untuk menginputkan string, karena lebih efisien dari scanf. Lalu terdapat looping for (b=0; a[b]; b++) maka if (a[b] >= '0' && a[b] <= '9') looping tersebut untuk menghitung jumlah angka, dimana kondisi awal c=0, jika memenuhi kondisi tersebut akan c++. Lalu untuk menghitung huruf besar dan kecil juga sama tinggal mengganti hurufnya saja. Setelah itu untuk tanda baca hanya f++ yang terletak pada else.

8.5.4 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 mengatur 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

```

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 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

```

➤ **Analisa**

Pada program tersebut berfungsi untuk mencetak teks yang diinputkan diulang terus menerus kebawah sebanyak karakter inputan. Pada source code saya menggunakan fgets untuk menyimpan string. lalu terdapat deklarasi tipe variabel integer untuk spasi, dan juga terdapat strlen untuk menghitung panjang string dari teks. Setelah itu looping untuk mencetak kolom dengan for (int i=0; i < strlen(teks); i++). lalu looping untuk baris, for (int j=0; j < spasi; j++). Setelah itu looping agar program tercetak seperti segitiga maka for (int k=0; k < a; k++) lalu spasi -- agar teks awal semakin berhurung dan a++ agar semakin bertambah.