

**LAPORAN PRAKTIKUM  
SISTEM OPERASI  
MODUL 10  
SIMULASI COMMAND**



**Disusun Oleh :**

**YESY LELY YESTIANA**

**L200210227**

**Kelas E**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS KOMUNIKASI DAN INFORMATIKA  
UNIVERSITAS MUHAMMADIYAH SURAKARTA  
TAHUN 2022/2023**

## Lembar Kerja Praktikum Modul 10

NIM	: L200210227	Nilai praktek	:
Nama	: Yesy Lely Yestiana		
Dosen Pengampu	: Heru Setiya N., ST, M.Kom	Tanda tangan	:
Nama Asisten	: -		
Tanggal Praktikum	: 20/12/2022		

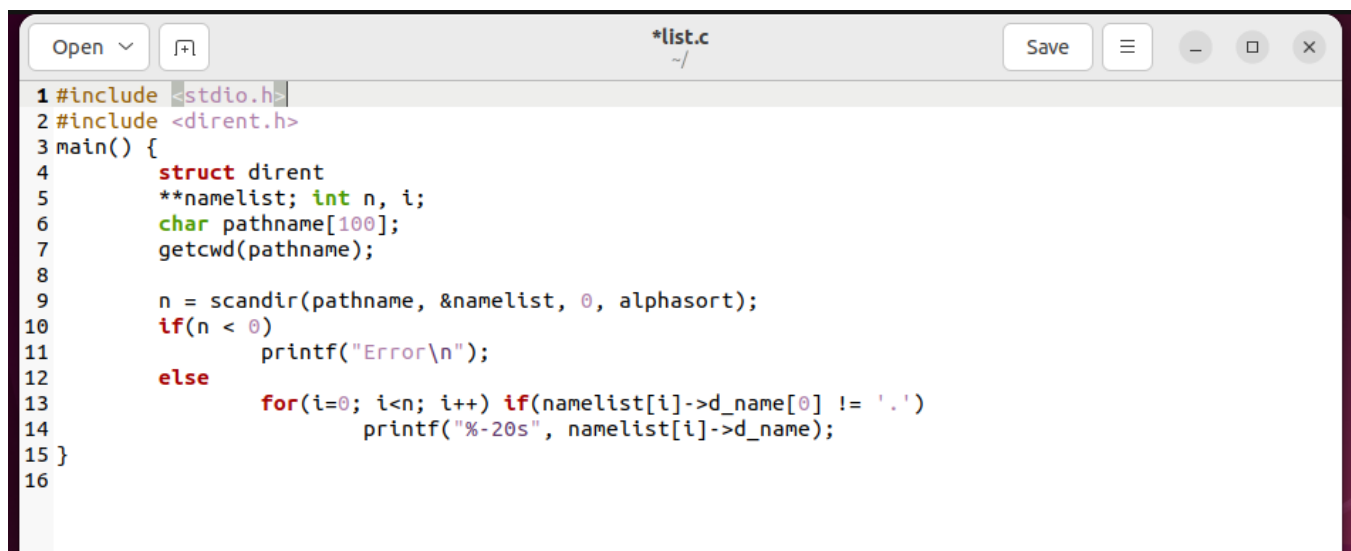
### Tugas!!!

#### ✓ Program untuk mensimulasi perintah 'ls'

Membuat kode program dengan algorithma sebagai berikut :

1. Menyimpan 'path' dari direktori kerja saat ini menggunakan perintah system call 'getcwd'.
2. Membaca isi direktori dari path di atas menggunakan perintah system call 'scandir' dan mengurutkan hasil pembacaannya dan menyimpannya dalam sebuah variabel array.
3. Menampilkan nama direktori (dname) dan nama file didalamnya jika file atau direktori tersebut tidak memiliki properti 'HIDE'.
4. Stop.

❖ Berikut merupakan screen shoot kode program :



```
1 #include <stdio.h>
2 #include <dirent.h>
3 main() {
4     struct dirent
5     **namelist; int n, i;
6     char pathname[100];
7     getcwd(pathname);
8
9     n = scandir(pathname, &namelist, 0, alphasort);
10    if(n < 0)
11        printf("Error\n");
12    else
13        for(i=0; i<n; i++) if(namelist[i]->d_name[0] != '.')
14            printf("%-20s", namelist[i]->d_name);
15 }
16
```

❖ Hasil output an :

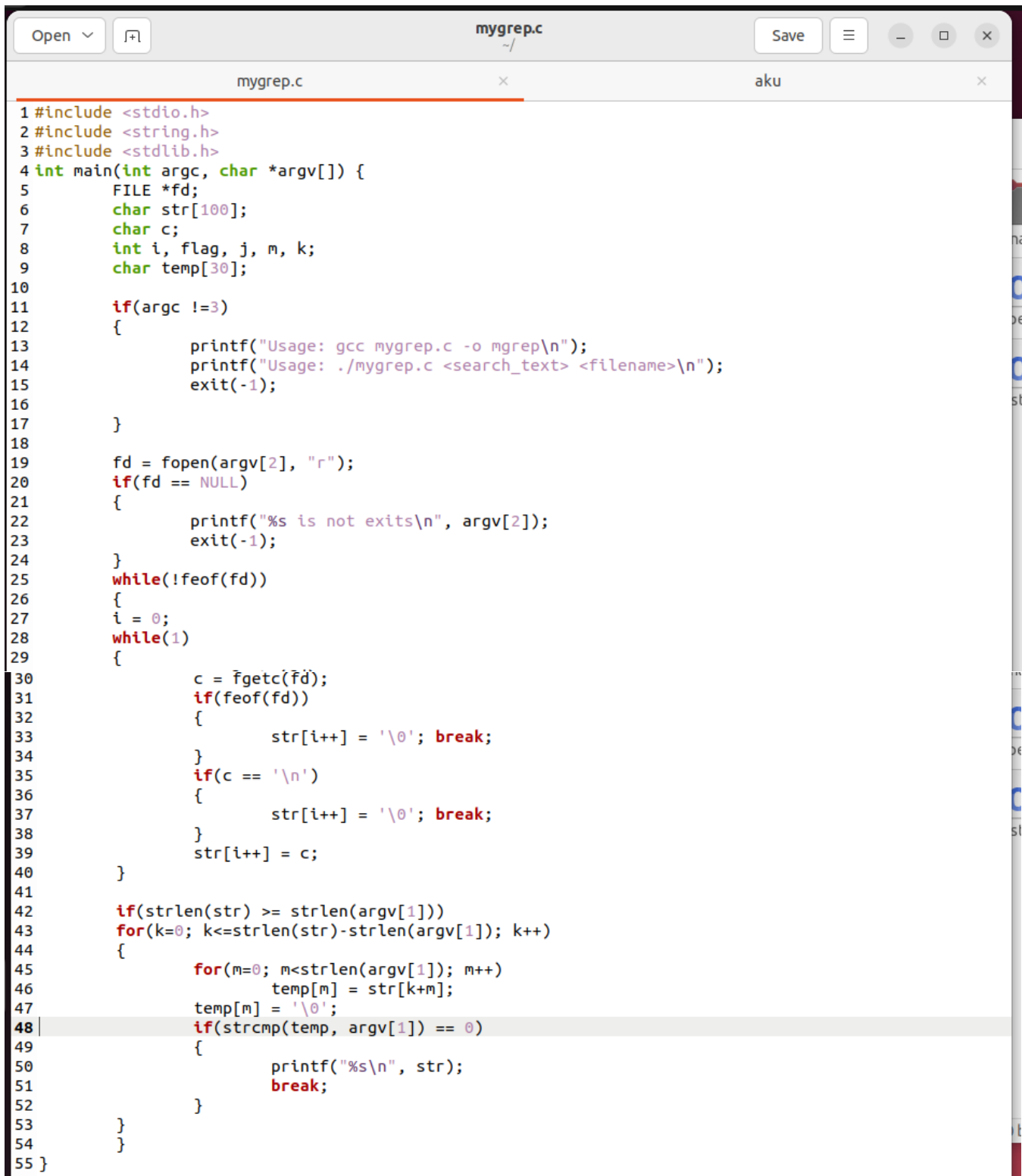
```
yesy@yesy-VirtualBox: ~  
yesy@yesy-VirtualBox:~$ gcc -o list list.c  
list.c: In function 'main':  
list.c:7:9: warning: implicit declaration of function 'getcwd'; did you mean 'getw'? [-Wimplicit-function-declaration]  
7 |         getcwd(pathname);  
  |         ^~~~~~  
  |         getw  
yesy@yesy-VirtualBox:~$ gcc -o list list.c  
list.c: In function 'main':  
list.c:7:9: warning: implicit declaration of function 'getcwd'; did you mean 'getw'? [-Wimplicit-function-declaration]  
7 |         getcwd(pathname);  
  |         ^~~~~~  
  |         getw  
yesy@yesy-VirtualBox:~$ ./list  
Desktop          Documents        Downloads        Music  
Pictures         Public           Templates        Videos  
a.out            dirlist.c       exec.c           fappend.c  
fcreate          fcreate.c       fork.c           fread.c  
list             list.c          snap             stat.c  
wait.c           yesy@yesy-VirtualBox:~$
```

### ✓ Program untuk mensimulasi perintah 'grep'

Membuat kode program dengan algoritma sebagai berikut :

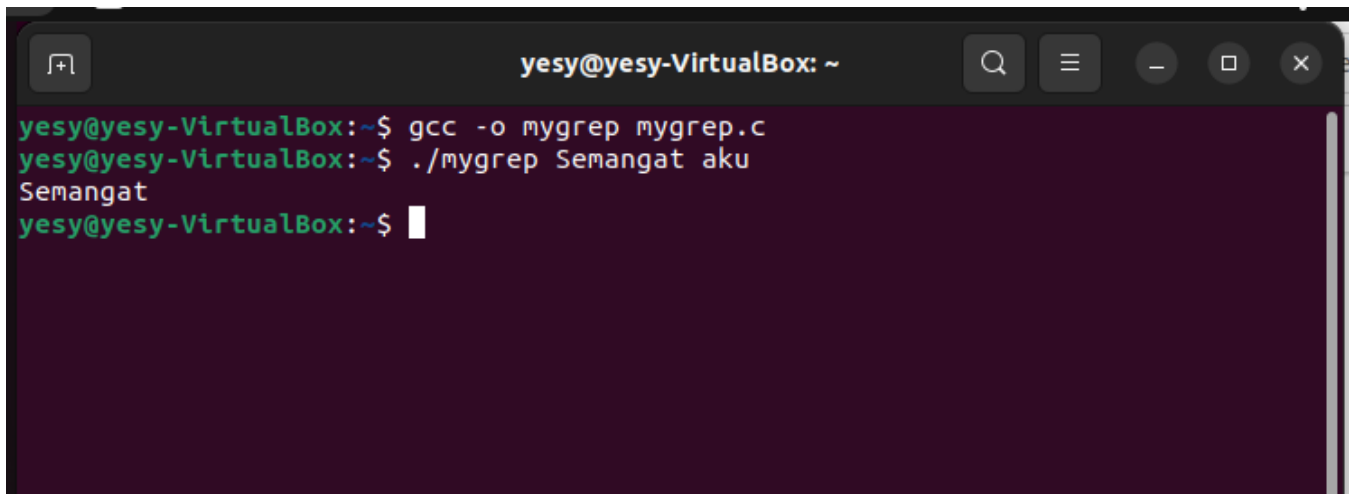
1. Gunakan nama file yang diberikan dalam argumen command-line.
2. Buka file dalam mode 'read-only' menggunakan perintah system call 'open'.
3. Jika file tidak ada, keluar program, stop.
4. Misal panjang string yang dicari adalah n.
5. Baca file perbaris sampai akhir file (END-OF-FILE), untuk setiap baris lakukan hal-hal berikut: (a) Periksa untuk mencari string dalam baris tersebut dengan dalam range 1-n, 2-n+1, dan seterusnya, (b) Jika string ditemukan tampilan baris tersebut di layar.
6. Tutup file menggunakan perintah 'close'.
7. Stop.

❖ Berikut merupakan screen shoot kode program :



```
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4 int main(int argc, char *argv[]) {
5     FILE *fd;
6     char str[100];
7     char c;
8     int i, flag, j, m, k;
9     char temp[30];
10
11     if(argc !=3)
12     {
13         printf("Usage: gcc mygrep.c -o mgrep\n");
14         printf("Usage: ./mygrep.c <search_text> <filename>\n");
15         exit(-1);
16     }
17
18     fd = fopen(argv[2], "r");
19     if(fd == NULL)
20     {
21         printf("%s is not exists\n", argv[2]);
22         exit(-1);
23     }
24     while(!feof(fd))
25     {
26         i = 0;
27         while(1)
28         {
29             c = fgetc(fd);
30             if(feof(fd))
31             {
32                 str[i++] = '\0'; break;
33             }
34             if(c == '\n')
35             {
36                 str[i++] = '\0'; break;
37             }
38             str[i++] = c;
39         }
40     }
41     if(strlen(str) >= strlen(argv[1]))
42     for(k=0; k<=strlen(str)-strlen(argv[1]); k++)
43     {
44         for(m=0; m<strlen(argv[1]); m++)
45             temp[m] = str[k+m];
46         temp[m] = '\0';
47         if(strcmp(temp, argv[1]) == 0)
48         {
49             printf("%s\n", str);
50             break;
51         }
52     }
53 }
54 }
55 }
```

❖ Hasil output an :



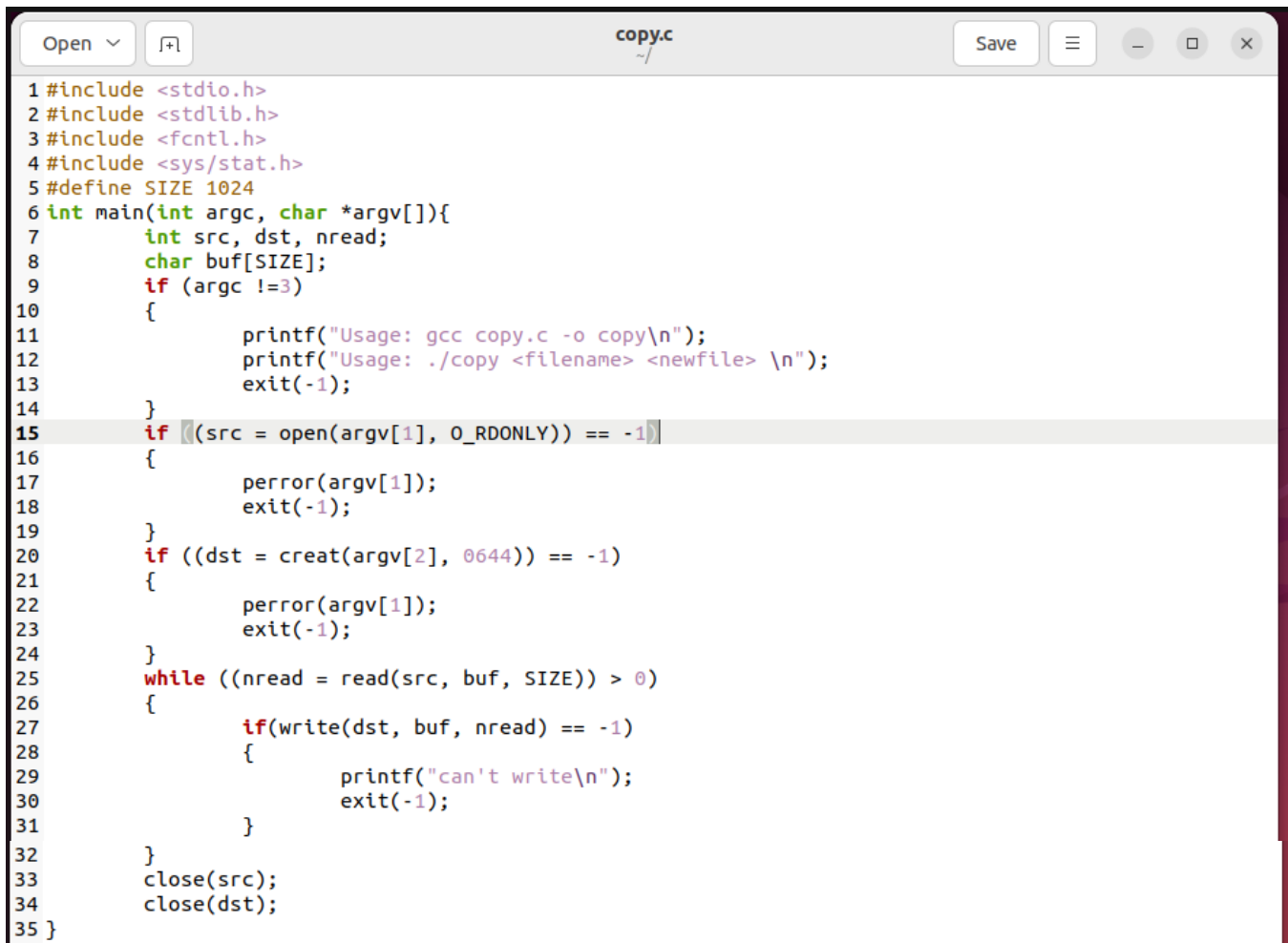
```
yesy@yesy-VirtualBox:~$ gcc -o mygrep mygrep.c
yesy@yesy-VirtualBox:~$ ./mygrep Semangat aku
Semangat
yesy@yesy-VirtualBox:~$
```

✓ **Program untuk mensimulasi perintah ‘cp’**

Membuat kode program dengan algoritma sebagai berikut :

1. Gunakan nama file untuk sumber dan tujuan dari argumen yang diberikan dalam command line.
2. Deklarasi sebuah buffer berukuran 1 KB.
3. Buka file sumber dalam mode ‘read-only’ menggunakan fungsi ‘open’.
4. Jika file sumber tidak ditemukan, stop keluar dari program.
5. Membuat file baru sebagai file target dengan menggunakan perintah ‘creat’.
6. Jika proses pembuatan file gagal, stop keluar dari program.
7. Proses penyalinan (copy) file dilakukan dengan cara berikut: (a) Membaca 1KB data dari file sumber dan menyimpan hasilnya dalam buffer menggunakan perintah ‘read’. (b) Menuliskan isi buffer dalam file target menggunakan perintah ‘write’. (c) Jika bertemu dengan kode ‘END-OF-FILE’ lanjut ke nomor 8, yang lain kembali ke perintah (a).
8. Tutup file sumber dan target menggunakan perintah ‘close’.
9. Stop.

❖ Berikut merupakan screen shoot kode program :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <fcntl.h>
4 #include <sys/stat.h>
5 #define SIZE 1024
6 int main(int argc, char *argv[]){
7     int src, dst, nread;
8     char buf[SIZE];
9     if (argc !=3)
10    {
11        printf("Usage: gcc copy.c -o copy\n");
12        printf("Usage: ./copy <filename> <newfile> \n");
13        exit(-1);
14    }
15    if ((src = open(argv[1], O_RDONLY)) == -1)
16    {
17        perror(argv[1]);
18        exit(-1);
19    }
20    if ((dst = creat(argv[2], 0644)) == -1)
21    {
22        perror(argv[1]);
23        exit(-1);
24    }
25    while ((nread = read(src, buf, SIZE)) > 0)
26    {
27        if(write(dst, buf, nread) == -1)
28        {
29            printf("can't write\n");
30            exit(-1);
31        }
32    }
33    close(src);
34    close(dst);
35 }
```

❖ Hasil output an :

```
yesy@yesy-VirtualBox: ~  
yesy@yesy-VirtualBox:~$ gcc copy.c -o copy  
copy.c: In function 'main':  
copy.c:25:25: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]  
25 |         while ((nread = read(src, buf, SIZE)) > 0)  
    |                        ^~~~~  
    |                        fread  
copy.c:27:20: warning: implicit declaration of function 'write'; did you mean 'fwrite'? [-Wimplicit-function-declaration]  
27 |         if(write(dst, buf, nread) == -1)  
    |            ^~~~~  
    |            fwrite  
copy.c:33:9: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]  
33 |         close(src);  
    |         ^~~~~  
    |         pclose  
yesy@yesy-VirtualBox:~$ ./copy aku saya  
yesy@yesy-VirtualBox:~$
```

### ✓ Program untuk mensimulasi perintah 'rm'

Membuat kode program dengan algoritma sebagai berikut :

1. Gunakan nama file yang diberikan dalam argumen command line.
2. Buka file dalam mode 'read-only' menggunakan perintah 'read'.
3. Jika file tidak ditemukan, stop keluar program.
4. Tutup file menggunakan perintah 'close'.
5. Menghapus file menggunakan perintah 'unlink'.
6. Stop.

❖ Berikut merupakan screen shoot kode program :

```
Open ▾ del.c ~/  
Save  
1 #include <stdio.h>  
2 #include <stdlib.h>  
3 #include <fcntl.h>  
4 int main(int argc, char *argv[]){  
5     int fd;  
6     if (argc !=2)  
7     {  
8         printf("Usage: gcc del.c -o del\n");  
9         printf("Usage: ./del <filename>\n");  
10        exit(-1);  
11    }  
12    fd = open(argv[1], O_RDONLY);  
13    if (fd !=-1)  
14    {  
15        close(fd);  
16        unlink(argv[1]);  
17    }  
18    else  
19        perror(argv[1]);  
20 }
```

❖ Hasil output an :

```
yesy@yesy-VirtualBox: ~  
yesy@yesy-VirtualBox:~$ gcc del.c -o del  
del.c: In function 'main':  
del.c:15:17: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]  
15 |         close(fd);  
    |         ^~~~~~  
    |         pclose  
del.c:16:17: warning: implicit declaration of function 'unlink' [-Wimplicit-function-declaration]  
16 |         unlink(argv[1]);  
    |         ^~~~~~  
yesy@yesy-VirtualBox:~$ ./del saya  
bash: ./del: No such file or directory  
yesy@yesy-VirtualBox:~$ ./del saya  
yesy@yesy-VirtualBox:~$
```