

**TUGAS**  
**PRAKTIKUM SISTEM OPERASI**  
**MODUL 8 SYSTEM CALL**



**Disusun Oleh :**  
**DIAN NUR HAYATI**  
**L200210229**  
**Kelas E**

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

NIM	: L200210229	Nilai :     Tanda tangan :
Nama	: Dian Nur Hayati	
Kelas	: E	
Dosen pengampu	: Heru Setiya Nugraha, ST, M.Kom	
Tanggal Praktikum	: 6 Desember 2022	

## Langkah Kerja

1. Membuat sebuah 'child process' (proses baru) dengan menggunakan system call 'fork'.

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 main() {
6     pid_t pid;
7     int x = 5;
8     pid = fork();
9     x++;
10    if (pid < 0)
11    {
12        printf("Process creation error"); exit(-1);
13    }
14    else if (pid == 0)
15    {
16        printf("Child process:");
17        printf("\nProcess id is %d", getpid());
18        printf("\nValue of x is %d", x);
19        printf("\nProcess id parent is %d\n\n", getppid());
20    }
21    else
22    {
23        printf("\nProcess id parent:");
24        printf("\nProcess id is %d", getpid());
25        printf("\nValue of x is %d", x);
26        printf("\nProcess id shell is %d\n", getppid());
27    }
28 }

```

Screenshot dari code fork.c

```

dian@dian-VirtualBox:~$ gcc 'fork.c'
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
5 | main() {
  | ~~~~~
dian@dian-VirtualBox:~$ ./a.out
Process id parent:
Process id is 32699
Value of x is 6
Process id shell is 27450
dian@dian-VirtualBox:~$ Child process:
Process id is 32700
Value of x is 6
Process id parent is 1245

```

Screenshot dari terminal fork.c

2. Menghentikan sementara (blocks) proses parent sampai dengan proses child selesai, menggunakan perintah system call 'wait'.

```
fork.c    x    wait.c    x    exec.c    x
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 #include <sys/wait.h>
6 main() {
7     int i, status;
8     pid_t pid;
9     pid = fork();
10
11     if (pid < 0){
12         printf("\nPembuatan proses gagal\n");
13         exit(-1);
14     }
15     else if(pid > 0)
16     {
17         wait(NULL);
18         printf ("\nParent starts\nNomor Genap:");
19         for (i=2;i<=10;i+=2)
20             printf ("%3d" ,i);
21         printf ("\nParent ends\n");
22     }
23     else if (pid == 0)
24     {
25         printf ("Child starts\nNomor Ganjil");
26         for (i=1;i<=10;i+=2)
27             printf ("%3d" ,i);
28         printf ("\nChild ends\n");
29     }
30 }
```

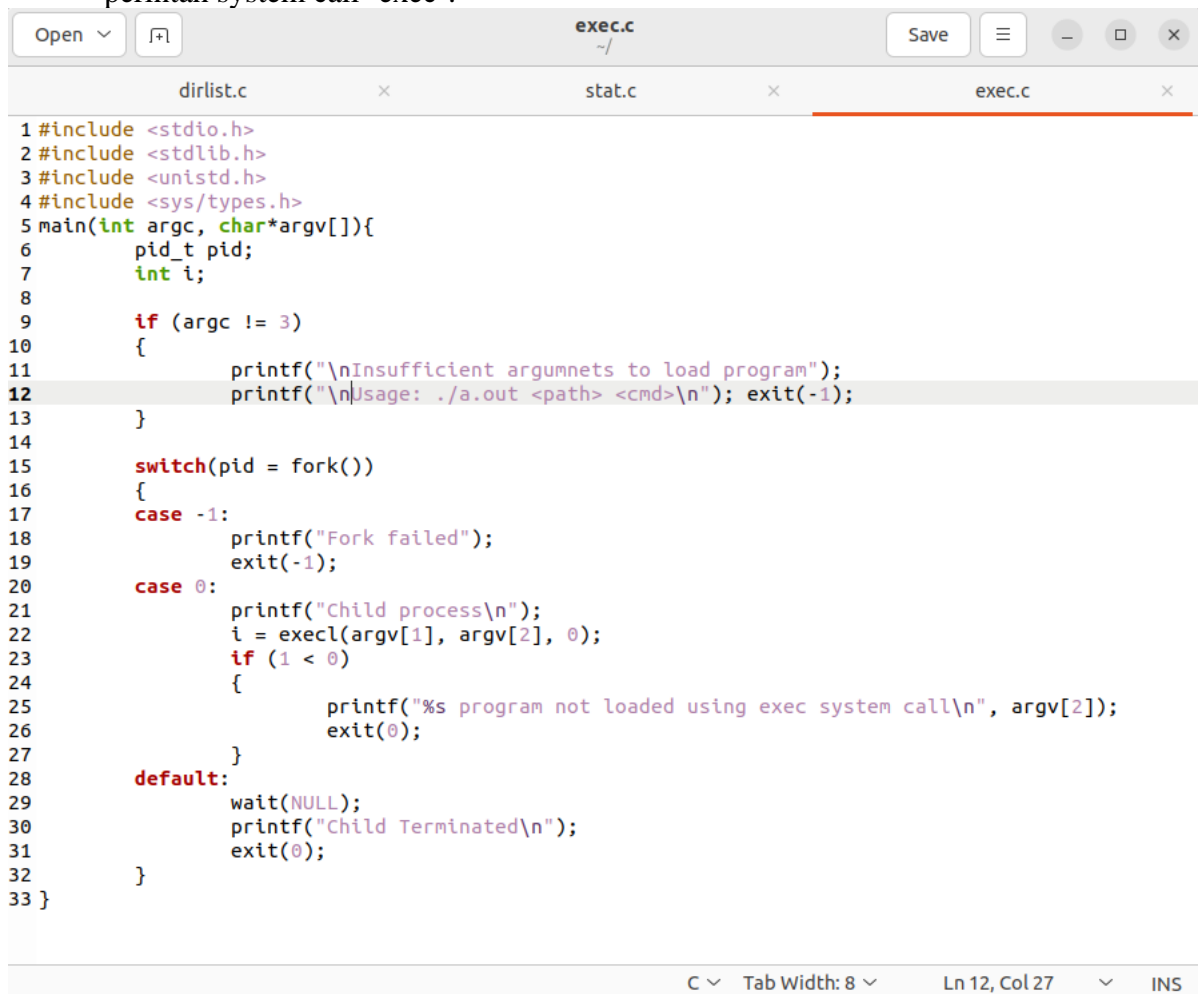
*Screenshot dari code wait.c*

```
dian@dian-VirtualBox:~$ gcc 'wait.c'
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
6 | main() {
  | ~~~~~
dian@dian-VirtualBox:~$ ./a.out
Child starts
Nomor Ganjil  1  3  5  7  9
Child ends

Parent starts
Nomor Genap:  2  4  6  8 10
Parent ends
dian@dian-VirtualBox:~$
```

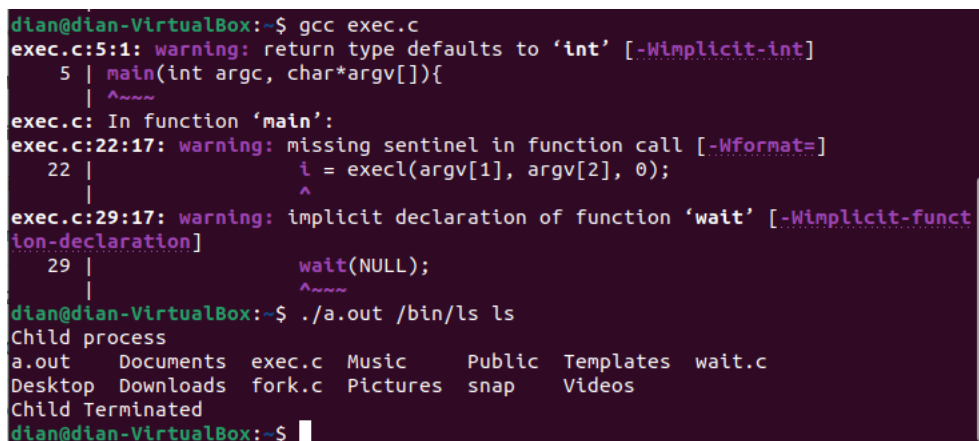
*Screenshot dari terminal wait.c*

3. Loading program yang dapat dieksekusi dalam sebuah 'child' proses menggunakan perintah system call 'exec'.



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 main(int argc, char*argv[]){
6     pid_t pid;
7     int i;
8
9     if (argc != 3)
10    {
11        printf("\nInsufficient argumnets to load program");
12        printf("\nUsage: ./a.out <path> <cmd>\n"); exit(-1);
13    }
14
15    switch(pid = fork())
16    {
17        case -1:
18            printf("Fork failed");
19            exit(-1);
20        case 0:
21            printf("Child process\n");
22            i = execl(argv[1], argv[2], 0);
23            if (1 < 0)
24            {
25                printf("%s program not loaded using exec system call\n", argv[2]);
26                exit(0);
27            }
28        default:
29            wait(NULL);
30            printf("Child Terminated\n");
31            exit(0);
32    }
33 }
```

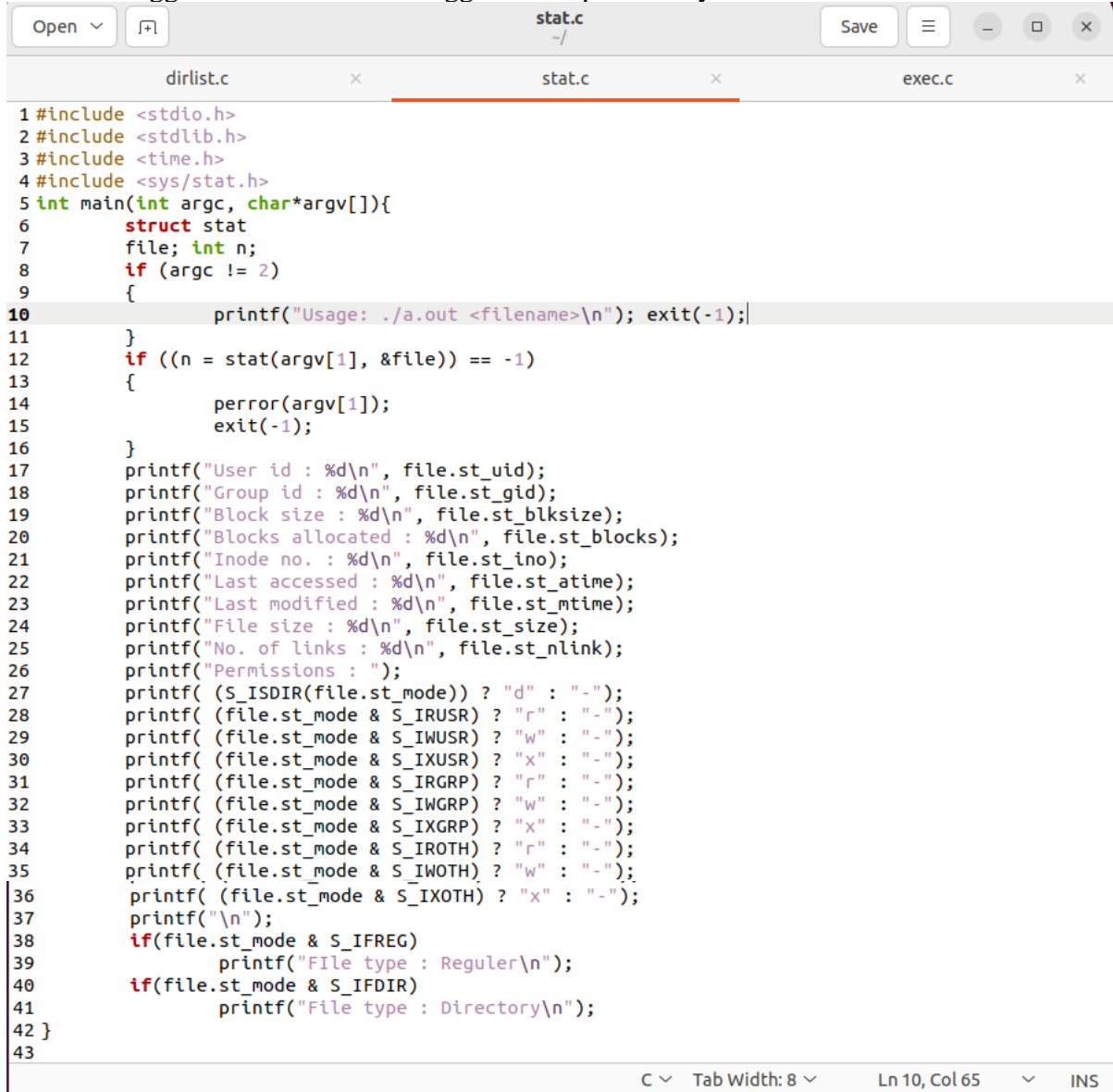
Screenshot dari code exec.c



```
dian@dian-VirtualBox:~$ gcc exec.c
exec.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
5 | main(int argc, char*argv[]){
  | ^~~~
exec.c: In function 'main':
exec.c:22:17: warning: missing sentinel in function call [-Wformat=]
22 |             i = execl(argv[1], argv[2], 0);
  |             ^
exec.c:29:17: warning: implicit declaration of function 'wait' [-Wimplicit-function-declaration]
29 |             wait(NULL);
  |             ^~~~
dian@dian-VirtualBox:~$ ./a.out /bin/ls ls
Child process
a.out  Documents  exec.c  Music  Public  Templates  wait.c
Desktop Downloads fork.c  Pictures snap  Videos
Child Terminated
dian@dian-VirtualBox:~$
```

Screenshot dari terminal exec.c

#### 4. Menggunakan status file menggunakan perintah system call 'stat'.



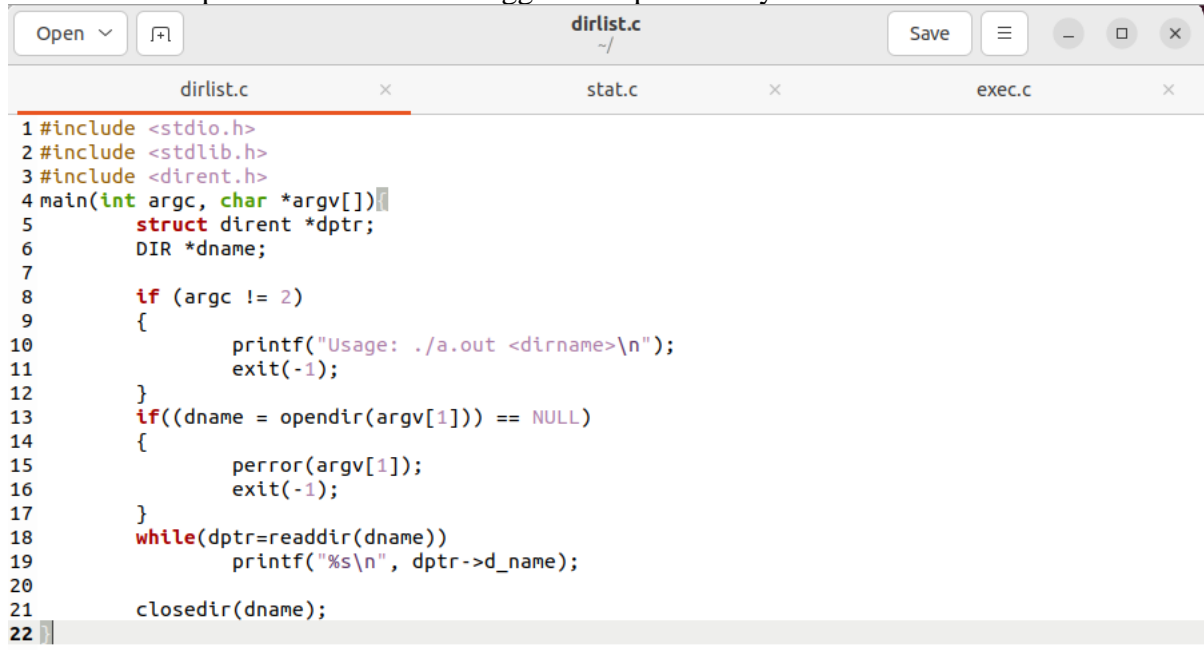
```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <time.h>
4 #include <sys/stat.h>
5 int main(int argc, char*argv[]){
6     struct stat
7     file; int n;
8     if (argc != 2)
9     {
10         printf("Usage: ./a.out <filename>\n"); exit(-1);
11     }
12     if ((n = stat(argv[1], &file)) == -1)
13     {
14         perror(argv[1]);
15         exit(-1);
16     }
17     printf("User id : %d\n", file.st_uid);
18     printf("Group id : %d\n", file.st_gid);
19     printf("Block size : %d\n", file.st_blksize);
20     printf("Blocks allocated : %d\n", file.st_blocks);
21     printf("Inode no. : %d\n", file.st_ino);
22     printf("Last accessed : %d\n", file.st_atime);
23     printf("Last modified : %d\n", file.st_mtime);
24     printf("File size : %d\n", file.st_size);
25     printf("No. of links : %d\n", file.st_nlink);
26     printf("Permissions : ");
27     printf( (S_ISDIR(file.st_mode)) ? "d" : "-");
28     printf( (file.st_mode & S_IRUSR) ? "r" : "-");
29     printf( (file.st_mode & S_IWUSR) ? "w" : "-");
30     printf( (file.st_mode & S_IXUSR) ? "x" : "-");
31     printf( (file.st_mode & S_IRGRP) ? "r" : "-");
32     printf( (file.st_mode & S_IWGRP) ? "w" : "-");
33     printf( (file.st_mode & S_IXGRP) ? "x" : "-");
34     printf( (file.st_mode & S_IROTH) ? "r" : "-");
35     printf( (file.st_mode & S_IWOTH) ? "w" : "-");
36     printf( (file.st_mode & S_IXOTH) ? "x" : "-");
37     printf("\n");
38     if(file.st_mode & S_IFREG)
39         printf("File type : Regular\n");
40     if(file.st_mode & S_IFDIR)
41         printf("File type : Directory\n");
42 }
43
```

Screenshot dari code stat.c

```
dian@dian-VirtualBox: ~  
dian@dian-VirtualBox:~$ gcc stat.c  
stat.c: In function 'main':  
stat.c:19:31: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blksize_t' {aka 'long int'} [-Wformat=]  
19 | printf("Block size : %d\n", file.st_blksize);  
   |                               ^~  
   |                               |  
   |                               int      __blksize_t {aka long int}  
   |                               %ld  
stat.c:20:37: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blkcnt_t' {aka 'long int'} [-Wformat=]  
20 | printf("Blocks allocated : %d\n", file.st_blocks);  
   |                               ^~  
   |                               |  
   |                               int      __blkcnt_t {aka long int}  
   |                               %ld  
stat.c:21:30: warning: format '%d' expects argument of type 'int', but argument 2 has type '__ino_t' {aka 'long unsigned int'} [-Wformat=]  
21 | printf("Inode no. : %d\n", file.st_ino);  
   |                               ^~  
   |                               |  
   |                               int      __ino_t {aka long unsigned int}  
   |                               %ld  
stat.c:22:34: warning: format '%d' expects argument of type 'int', but argument 2 has type '__time_t' {aka 'long int'} [-Wformat=]  
22 | printf("Last accessed : %d\n", file.st_atime);  
   |                               ^~  
   |                               |  
   |                               int      __time_t {aka long int}  
   |                               %ld  
stat.c:23:34: warning: format '%d' expects argument of type 'int', but argument 2 has type '__time_t' {aka 'long int'} [-Wformat=]  
23 | printf("Last modified : %d\n", file.st_mtime);  
   |                               ^~  
   |                               |  
   |                               int      __time_t {aka long int}  
   |                               %ld  
stat.c:24:30: warning: format '%d' expects argument of type 'int', but argument 2 has type '__off_t' {aka 'long int'} [-Wformat=]  
24 | printf("File size : %d\n", file.st_size);  
   |                               ^~  
   |                               |  
   |                               int      __off_t {aka long int}  
   |                               %ld  
stat.c:25:33: warning: format '%d' expects argument of type 'int', but argument 2 has type '__nlink_t' {aka 'long unsigned int'} [-Wformat=]  
25 | printf("No. of links : %d\n", file.st_nlink);  
   |                               ^~  
   |                               |  
   |                               int      __nlink_t {aka long unsigned int}  
   |                               %ld  
dian@dian-VirtualBox:~$ ./a.out stat.c  
User id : 1000  
Group id : 1000  
Block size : 4096  
Blocks allocated : 8  
Inode no. : 929111  
Last accessed : 1670839554  
Last modified : 1670839548  
File size : 1350  
No. of links : 1  
Permissions : -rw-rw-r--  
File type : Regular  
dian@dian-VirtualBox:~$
```

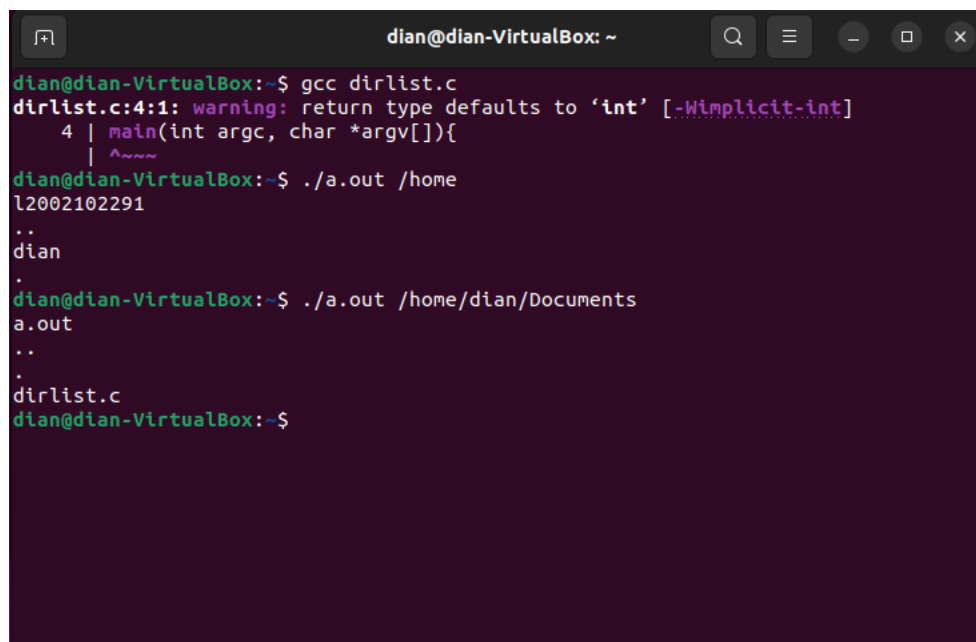
Screenshot dari terminal stat.c

5. Menampilkan isi direktori menggunakan perintah system call 'readdir'.



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <dirent.h>
4 main(int argc, char *argv[])
5 {
6     struct dirent *dptr;
7     DIR *dname;
8
9     if (argc != 2)
10    {
11        printf("Usage: ./a.out <dirname>\n");
12        exit(-1);
13    }
14    if((dname = opendir(argv[1])) == NULL)
15    {
16        perror(argv[1]);
17        exit(-1);
18    }
19    while(dptr=readdir(dname))
20        printf("%s\n", dptr->d_name);
21    closedir(dname);
22 }
```

*Screenshot dari code dirlist.c*



```
dian@dian-VirtualBox: ~
dian@dian-VirtualBox:~$ gcc dirlist.c
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]
  4 | main(int argc, char *argv[])
    | ^~~~~
dian@dian-VirtualBox:~$ ./a.out /home
l2002102291
..
dian
.
dian@dian-VirtualBox:~$ ./a.out /home/dian/Documents
a.out
..
.
dirlist.c
dian@dian-VirtualBox:~$
```

*Screenshot dari terminal dirlist.c*