

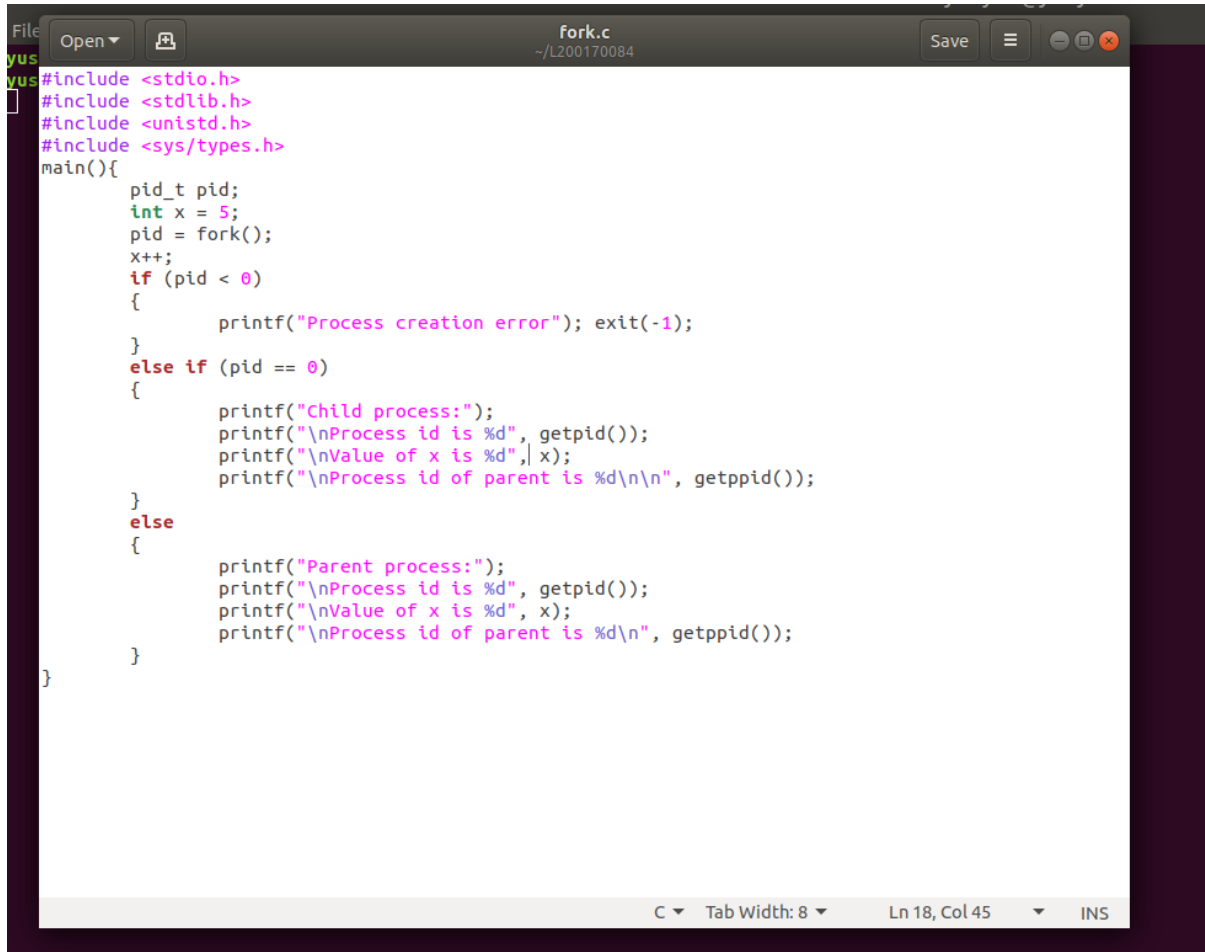
Nama : Malvin Iqbal Firdaus

NIM : L200170035

Kelas : B

MODUL 08

1. Ketikkan 'gedit fork.c' kemudian tulis kode program seperti di bawah ini:

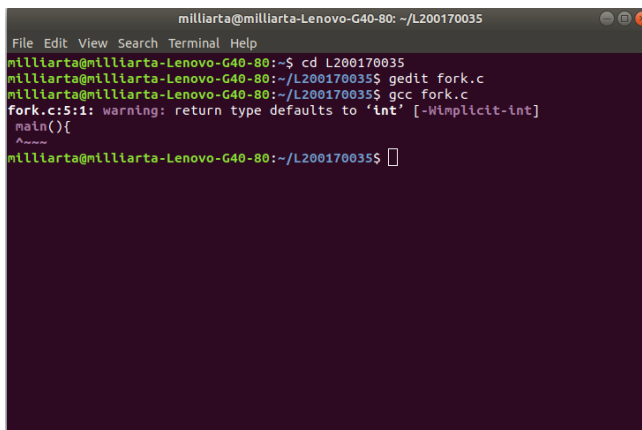


```
File Open [icon] fork.c ~/L200170084 Save [icon] [icon] [icon] [icon]
yus
yus
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
main(){
    pid_t pid;
    int x = 5;
    pid = fork();
    x++;
    if (pid < 0)
    {
        printf("Process creation error"); exit(-1);
    }
    else if (pid == 0)
    {
        printf("Child process:");
        printf("\nProcess id is %d", getpid());
        printf("\nValue of x is %d", x);
        printf("\nProcess id of parent is %d\n\n", getppid());
    }
    else
    {
        printf("Parent process:");
        printf("\nProcess id is %d", getpid());
        printf("\nValue of x is %d", x);
        printf("\nProcess id of parent is %d\n", getppid());
    }
}
```

C Tab Width: 8 Ln 18, Col 45 INS

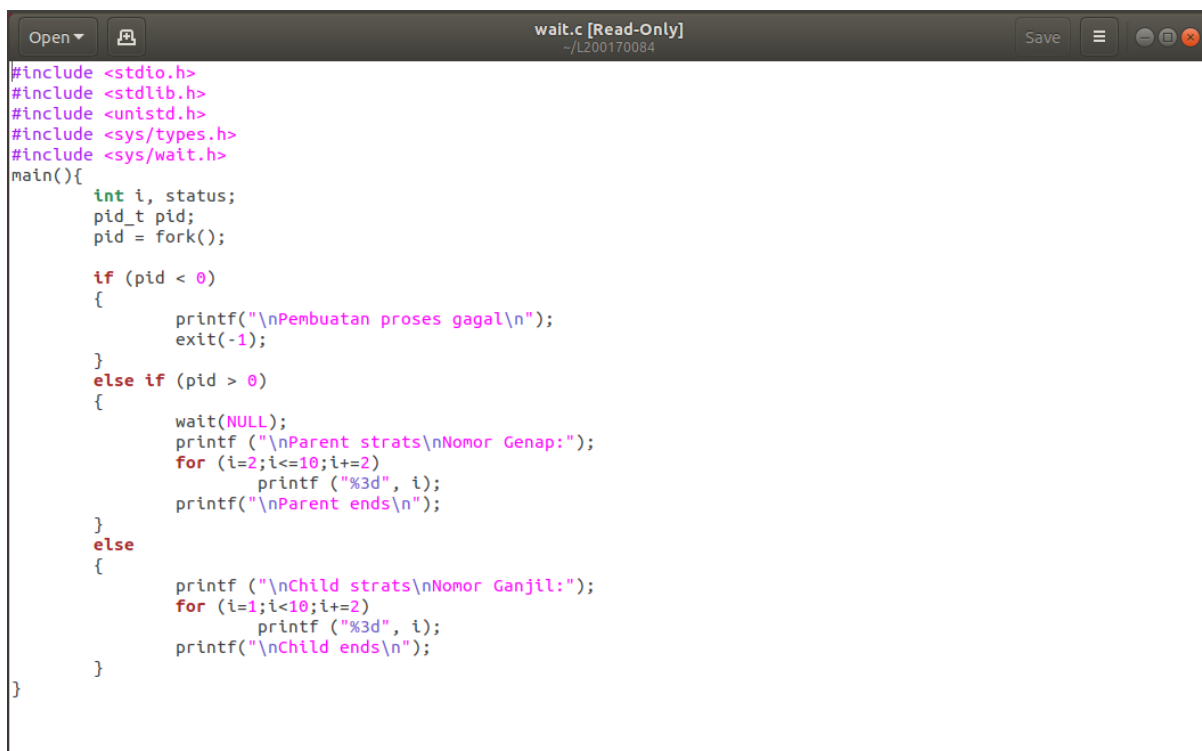
2. Kemudian ngengkompilasi dan menjalankan program tersebut:

- \$gcc fork.c
- \$./a.out



```
milliarta@milliarta-Lenovo-G40-80: ~/L200170035
File Edit View Search Terminal Help
milliarta@milliarta-Lenovo-G40-80:~$ cd L200170035
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit fork.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc fork.c
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(){
^~~~~~
milliarta@milliarta-Lenovo-G40-80:~/L200170035$
```

3. Ketikkan 'gedit wait.c' kemudian tulis kode program seperti di bawah ini:



```
wait.c [Read-Only]
~/L200170084
Open Save
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
main(){
    int i, status;
    pid_t pid;
    pid = fork();

    if (pid < 0)
    {
        printf("\nPembuatan proses gagal\n");
        exit(-1);
    }
    else if (pid > 0)
    {
        wait(NULL);
        printf ("\nParent strats\nNomor Genap:");
        for (i=2;i<=10;i+=2)
            printf ("%3d", i);
        printf("\nParent ends\n");
    }
    else
    {
        printf ("\nChild strats\nNomor Ganjil:");
        for (i=1;i<10;i+=2)
            printf ("%3d", i);
        printf("\nChild ends\n");
    }
}
```

4. Kemudian ngengkompilasi dan menjalankan program tersebut:

- \$gcc wait.c
- \$./a.out

```
milliarta@milliarta-Lenovo-G40-80: ~/L200170035
File Edit View Search Terminal Help
Value of x is 6
Process id of parent if 3352

milliarta@milliarta-Lenovo-G40-80:~/L200170035$ child process:
child: command not found
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ Child process
Child: command not found
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ Child process:
Child: command not found
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit exec.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit wait.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc wait.c
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(){
^~~~~~
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ ./a.out
Child starts
Nomor Ganjil: 1 3 5 7 9
Child ends

Parent starts
Nomor Genap: 2 4 6 8 10
Parent ends
milliarta@milliarta-Lenovo-G40-80:~/L200170035$
```

5. Lanjutkan dengan mengetikkan 'gedit exec.c' kemudian tulis kode program seperti di bawah ini:

```
exec.c [Read-Only]
~/L200170084
Save

#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
main(int argc, char*argv[]){
    pid_t pid;
    int i;

    if (argc !=3)
    {
        printf("\nInsufficient arguments to load program");
        printf("\nUsage: ./a.out <path> <cmd>\n"); exit(-1);
    }

    switch(pid = fork())
    {
        case -1:
            printf("Fork failed");
            exit(-1);
        case 0:
            printf("Child process\n");
            i = execl(argv[1], argv[2], 0);
            if (i < 0)
            {
                printf("%s program not loaded using exec system call\n",
argv[2]);
                exit(-1);
            }
        default:
            wait(NULL);
            printf("Child Terminated\n");
            exit(0);
    }
}
```

6. Kemudian ngengkompilasi dan menjalankan program tersebut:

- \$gcc exec.c
- \$./a.out /bin/ls ls

```
milliarta@milliarta-Lenovo-G40-80: ~/L200170035
File Edit View Search Terminal Help
Nomor Genap: 2 4 6 8 10
Parent ends
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit exec.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc exex.c
gcc: error: exex.c: No such file or directory
gcc: fatal error: no input files
compilation terminated.
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc exec.c
exec.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char*argv[]){
^
exec.c: In function 'main':
exec.c:23:3: warning: missing sentinel in function call [-Wformat=]
    i = execl(argv[1], argv[2], 0);
    ^
exec.c:30:3: warning: implicit declaration of function 'wait'; did you mean 'main'? [-Wimplicit-function-declaration]
    wait(NULL);
    ^
main
exec.c:16:2: warning: switch condition has boolean value [-Wswitch-bool]
    switch(pid == fork())
    ^
milliarta@milliarta-Lenovo-G40-80:~/L200170035$
```

7. Lanjutkan dengan mengetikkan 'gedit stat.c' kemudian tulis kode program seperti di bawah ini:

```
stat.c [Read-Only]
~/L200170084
Save

#include <stdio.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <time.h>

int main(int argc, char*argv[]){
    struct stat
    file; int n;
    if (argc != 2)
    {
        printf("Usage: ./a.out <filename>\n"); exit(-1);
    }
    if ((n = stat(argv[1], &file)) == -1)
    {
        perror(argv[1]);
        exit(-1);
    }
    printf("User id : %d\n", file.st_uid);
    printf("Group id : %d\n", file.st_gid);
    printf("Block size : %d\n", file.st_blksize);
    printf("Blocks allocated : %d\n", file.st_blocks);
    printf("Inode no. : %d\n", file.st_ino);
    printf("Last accessed : %s\n", ctime(&(file.st_atime)));
    printf("Last modifier : %s\n", ctime(&(file.st_mtime)));
    printf("File size : %d bytes\n", file.st_size);
    printf("No. of links : %d\n", file.st_nlink);
    printf("Permissions : ");
    printf( (S_ISDIR(file.st_mode)) ? "d" : "-");
    printf( (file.st_mode & S_IRUSR) ? "r" : "-");
    printf( (file.st_mode & S_IWUSR) ? "w" : "-");
    printf( (file.st_mode & S_IXUSR) ? "x" : "-");
    printf( (file.st_mode & S_IRGRP) ? "r" : "-");
    printf( (file.st_mode & S_IWGRP) ? "w" : "-");
    printf( (file.st_mode & S_IXGRP) ? "x" : "-");
    printf( (file.st_mode & S_IROTH) ? "r" : "-");
    printf( (file.st_mode & S_IWOTH) ? "w" : "-");
    printf( (file.st_mode & S_IXOTH) ? "x" : "-");
    printf("\n");
    if(file.st_mode & S_IFREG)
        printf("File type : Regular\n");
    if(file.st_mode & S_IFDIR)
        printf("File type : Directory\n");
}
```

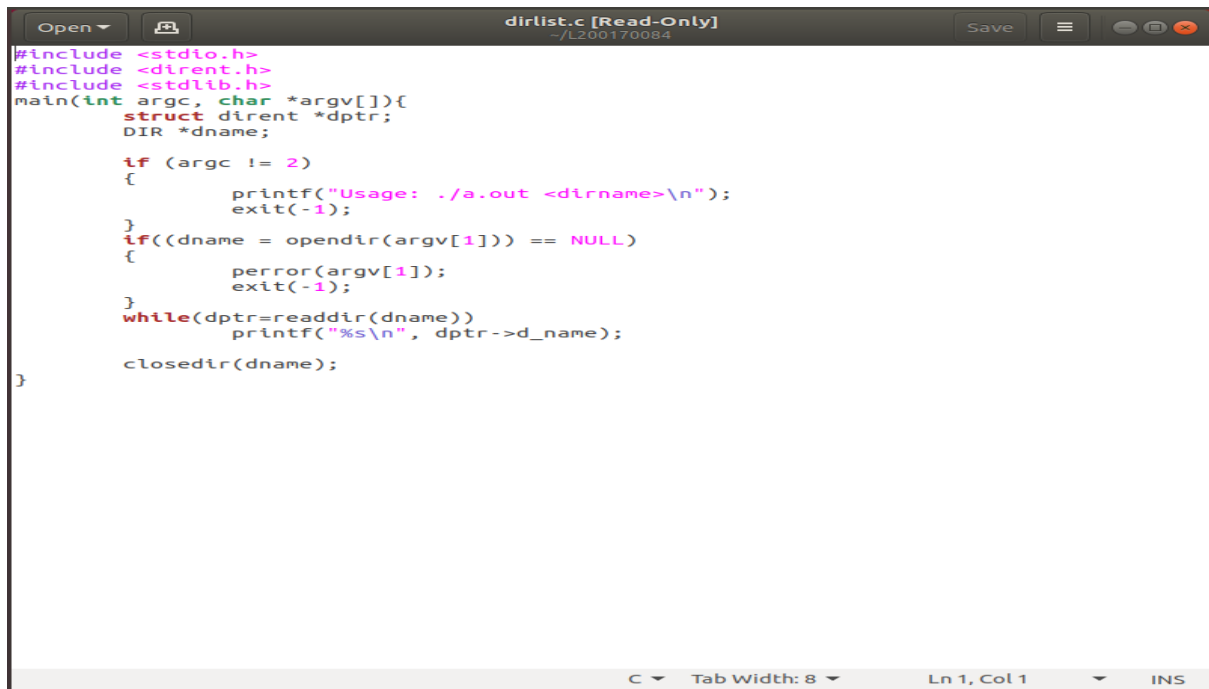
8. Kemudian ngengkompilasi dan menjalankan program tersebut:

- \$gcc stat.c
- \$./a.out /bin/ls

A screenshot of a Linux terminal window titled "Terminal". The terminal shows the following sequence of events:

- The user runs `exec.c:16:2: warning: switch condition has boolean value [-Wswitch-bool]`.
- The user runs `milliarta@milliarta-Lenovo-G40-80: ~/L200170035$./a.out /bin/lsls`.
- The output shows an error: `Insufficient arguments to load program Usage: ./a.out <path> <cmd>`.
- The user runs `milliarta@milliarta-Lenovo-G40-80:~/L200170035$./a.out /bin/lsls /ls`.
- The output shows a child process being terminated.
- The user runs `a.out ./dirlist.c exec.c fork.c stat.c wait.c`.
- The user runs `milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc stat.c`.
- The user runs `milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit stat.c`.
- The user runs `milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc stat.c`.
- The user runs `stat.c: In function 'main':`.
- The user runs `stat.c:19:24: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blksize_t {aka long int}' [-Wformat=]`.
- The user runs `printf("Block size : %d\n", file.st_blksize);`.
- The user runs `%ld`.
- The user runs `stat.c:20:30: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blkcnt_t {aka long int}' [-Wformat=]`.
- The user runs `printf("Blocks allocated : %d\n", file.st_blocks);`.
- The user runs `%ld`.
- The user runs `stat.c:21:23: warning: format '%d' expects argument of type 'int', but argument 2 has type '__ino_t {aka long unsigned int}' [-Wformat=]`.
- The user runs `printf("Inode no. : %d\n", file.st_ino);`.
- The user runs `%ld`.
- The user runs `stat.c:24:23: warning: format '%d' expects argument of type 'int', but argument 2 has type '__off_t {aka long int}' [-Wformat=]`.
- The user runs `printf("File size : %d bytes\n", file.st_size);`.
- The user runs `%ld`.
- The user runs `stat.c:25:26: warning: format '%d' expects argument of type 'int', but argument 2 has type '__nlink_t {aka long unsigned int}' [-Wformat=]`.
- The user runs `printf("No. of links : %d\n", file.st_nlink);`.
- The user runs `%ld`.
- The user runs `milliarta@milliarta-Lenovo-G40-80:~/L200170035$`.

9. Lanjutkan dengan mengetikkan 'gedit dirlist.c' kemudian tulis kode program seperti di bawah ini:



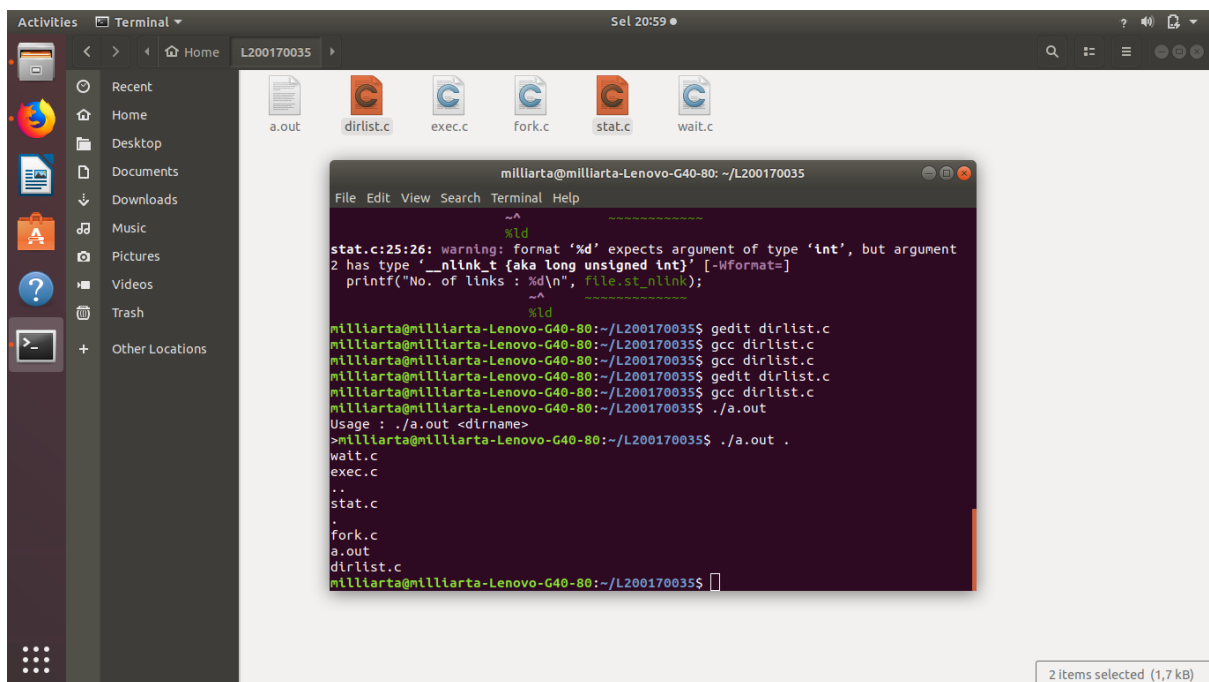
```
dirlist.c [Read-Only]
~/L200170084

#include <stdio.h>
#include <dirent.h>
#include <stdlib.h>
main(int argc, char *argv[]){
    struct dirent *dptr;
    DIR *dname;

    if (argc != 2)
    {
        printf("Usage: ./a.out <dirname>\n");
        exit(-1);
    }
    if((dname = opendir(argv[1])) == NULL)
    {
        perror(argv[1]);
        exit(-1);
    }
    while(dptr=readdir(dname))
        printf("%s\n", dptr->d_name);
    closedir(dname);
}
```

10. Kemudian ngengkompilasi dan menjalankan program tersebut:

- \$gcc dirlist.c
- \$./a.out .



Activities Terminal ▾ Sel 20:59 ●

L200170035

a.out dirlist.c exec.c fork.c stat.c wait.c

```
milliarta@milliarta-Lenovo-G40-80: ~/L200170035
File Edit View Search Terminal Help

stat.c:25:26: warning: format '%d' expects argument of type 'int', but argument
2 has type '_nlink_t {aka long unsigned int}' [-Wformat=]
printf("No. of links : %d\n", file.st_nlink);
                        ^~
                        %ld

milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gedit dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ gcc dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$ ./a.out
Usage : ./a.out <dirname>
>milliarta@milliarta-Lenovo-G40-80:~/L200170035$ ./a.out .
wait.c
exec.c
..
stat.c
.
fork.c
a.out
dirlist.c
milliarta@milliarta-Lenovo-G40-80:~/L200170035$
```

2 items selected (1,7 kB)