

Nama : Ismi Dzikrina

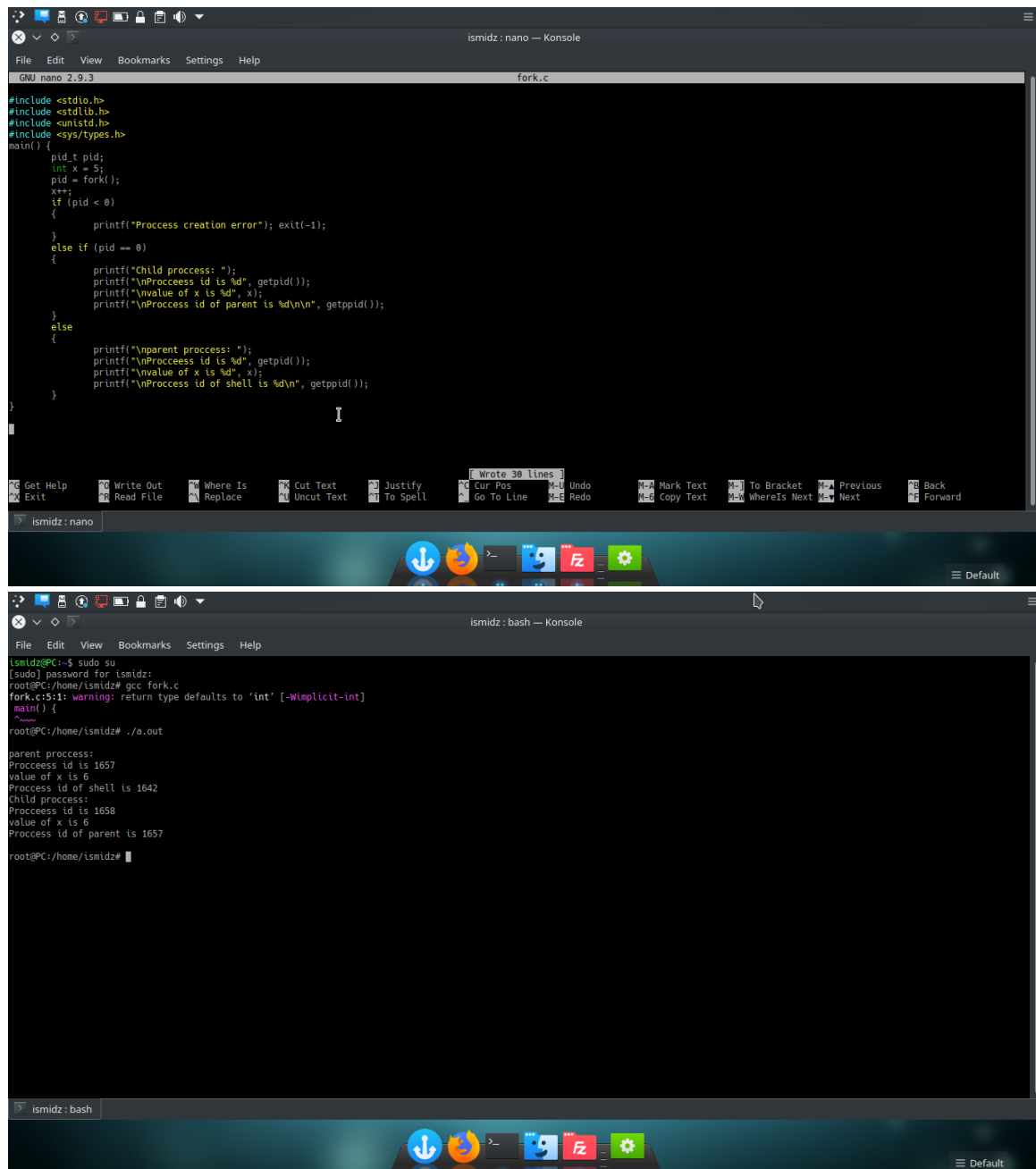
NIM : L200180010

Kelas : A

## MODUL 8

### SYSTEM CALL

#### 1. Fork.c



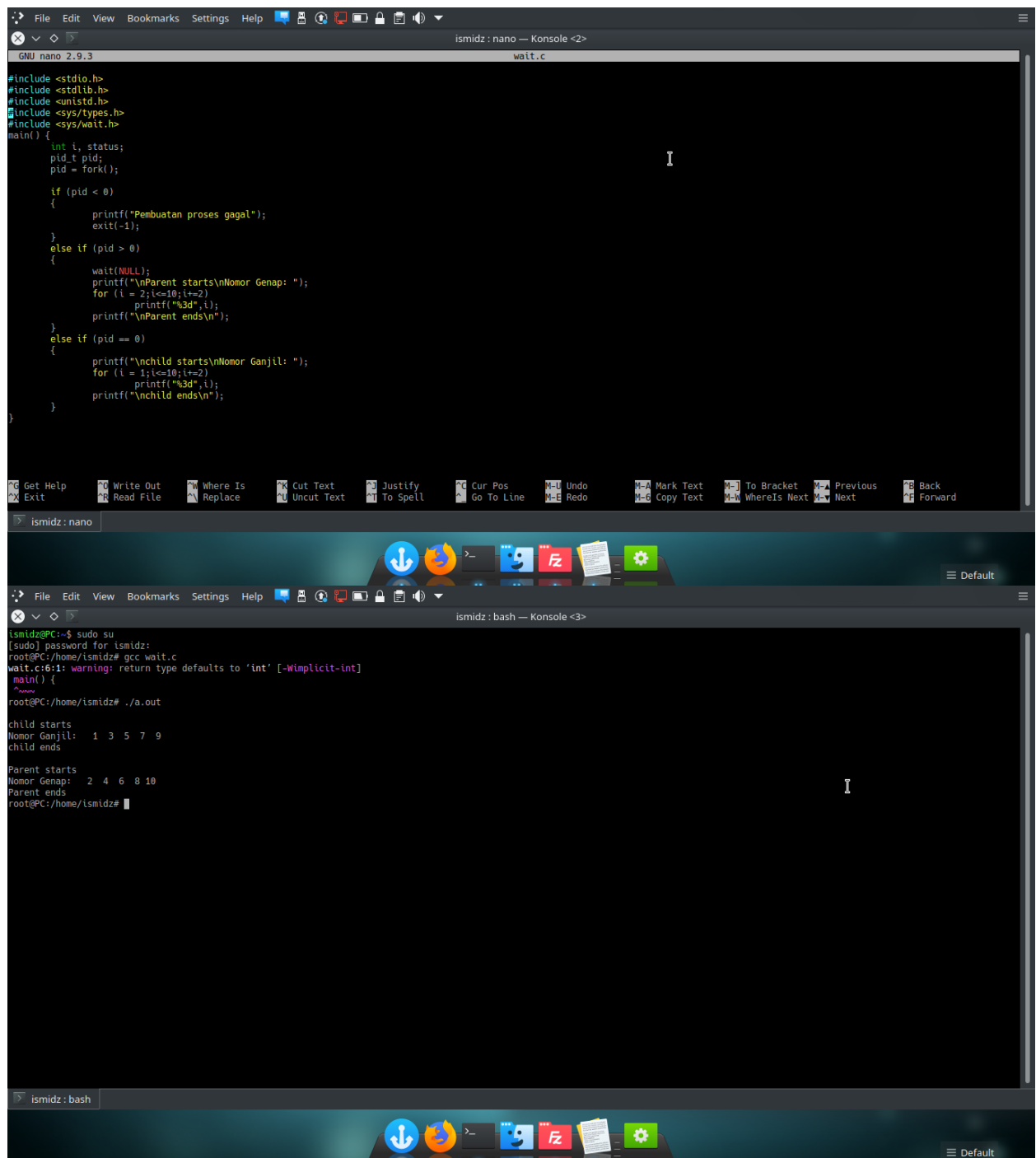
The first screenshot shows a nano text editor window titled 'ismidz: nano — Konsole' editing a file named 'fork.c'. The code is as follows:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
main() {
    pid_t pid;
    int x = 5;
    pid = fork();
    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", getppid());
    }
    else
    {
        printf("\nparent process: ");
        printf("\nProcess id is %d", getpid());
        printf("\nvalue of x is %d", x);
        printf("\nProcess id of shell is %d\n", getppid());
    }
}
```

The second screenshot shows a terminal window titled 'ismidz: bash — Konsole' where the program is compiled and executed. The commands and output are:

```
ismidz@PC:~$ sudo su
[sudo] password for ismidz:
root@PC:~/ismidz# gcc fork.c
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main() {
^
root@PC:~/ismidz# ./a.out
parent process:
Process id is 1657
value of x is 6
Process id of shell is 1642
Child process:
Process id is 1658
value of x is 6
Process id of parent is 1657
root@PC:~/ismidz#
```

## 2. Wait.c



```
GNU nano 2.9.3
ismidz: nano — Konsole <2>
wait.c

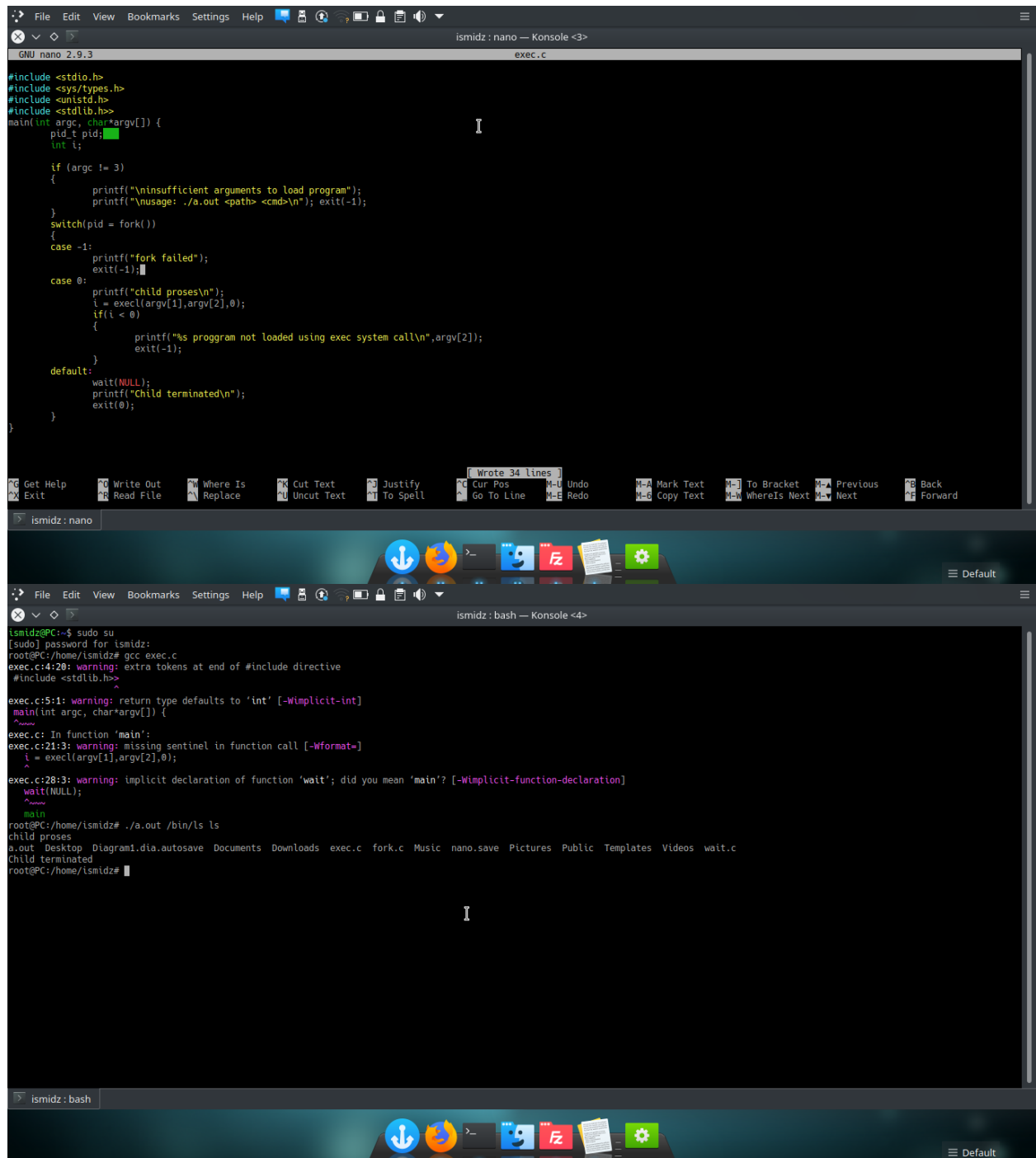
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
main() {
    int i, status;
    pid_t pid;
    pid = fork();
    if (pid < 0)
    {
        printf("Pembuatan proses gagal");
        exit(-1);
    }
    else if (pid > 0)
    {
        wait(NULL);
        printf("\nParent starts\nNomor Genap: ");
        for (i = 2; i <= 10; i += 2)
            printf("%3d", i);
        printf("\nParent ends\n");
    }
    else if (pid == 0)
    {
        printf("\nchild starts\nNomor Ganjil: ");
        for (i = 1; i <= 10; i += 2)
            printf("%3d", i);
        printf("\nchild ends\n");
    }
}

ismidz: nano

ismidz: bash — Konsole <3>
ismidz@PC:~$ sudo su
[sudo] password for ismidz:
root@PC:/home/ismidz# gcc wait.c
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
main() {
^~~~~~
root@PC:/home/ismidz# ./a.out
child starts
Nomor Ganjil:  1  3  5  7  9
child ends

Parent starts
Nomor Genap:  2  4  6  8 10
Parent ends
root@PC:/home/ismidz#
```

### 3. Exec.c



```
isimidz: nano - Konsole <3>
GNU nano 2.9.3
exec.c

#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 proses\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);
    }
}

isimidz: nano
Wrote 34 lines
Exit Get Help Write Out Read File Where Is Replace Cut Text Uncut Text Justify To Spell Cur Pos Go To Line Undo Redo Mark Text Copy Text To Bracket WhereIs Next Previous Next Back Forward

isimidz: bash - Konsole <4>
isimidz@PC:~$ sudo su
[sudo] password for isimidz:
root@PC:/home/isimidz# gcc exec.c
exec.c:4:20: warning: extra tokens at end of #include directive
#include <stdlib.h>
^
exec.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char*argv[]) {
^~~~~
exec.c: In function 'main':
exec.c:21:3: warning: missing sentinel in function call [-Wformat=]
    i = execl(argv[1],argv[2],0);
    ^
exec.c:28:3: warning: implicit declaration of function 'wait'; did you mean 'main'? [-Wimplicit-function-declaration]
    wait(NULL);
    ^~~~~
exec.c:29:1: warning: 'main' defined but not used [-Wunused-function]
main
^
root@PC:/home/isimidz# ./a.out /bin/ls ls
child proses
a.out Desktop Diagram1.dia.autosave Documents Downloads exec.c fork.c Music nano.save Pictures Public Templates Videos wait.c
Child terminated
root@PC:/home/isimidz#
```

```
File Edit View Bookmarks Settings Help
GNU nano 2.9.3 stat.c

#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 : %ld\n", file.st_blksize);
    printf("Blocks allocated : %ld\n", file.st_blocks);
    printf("Inode no. : %ld\n", file.st_ino);
    printf("Last accessed : %s\n", ctime(&file.st_atime));
    printf("Last modified : %s\n", ctime(&file.st_mtime));
    printf("File size : %ld bytes\n", file.st_size);
    printf("No. of link : %ld\n", file.st_nlink);
    printf("Permissions :");
    printf(" (%S_IROTH(file.st_mode)) ? 'd' : '-');
    printf(" (%S_IRUSR) ? 'r' : '-');
    printf(" (%S_IWUSR) ? 'w' : '-');
    printf(" (%S_IXUSR) ? 'x' : '-');
    printf(" (%S_IRGRP) ? 'r' : '-');
    printf(" (%S_IWGRP) ? 'w' : '-');
    printf(" (%S_IXGRP) ? 'x' : '-');
    printf(" (%S_IROTH) ? 'r' : '-');
    printf(" (%S_IWOTH) ? 'w' : '-');
    printf(" (%S_IXOTH) ? 'x' : '-');
    printf("\n");
    if(file.st_mode & S_IFREG)
        printf("File type : Regular\n");
    if(file.st_mode & S_IFDIR)

Get Help Write Out Where Is Cut Text Justify Cur Pos Undo Mark Text To Bracket Previous Back
Exit Read File Replace Unlink To Spell Share Copy Text WhereIs Next Next Forward

ismidz: nano ismidz: bash ismidz: nano ismidz: bash

File Edit View Bookmarks Settings Help
ismidz: bash — Konsole

ismidz@PC:~$ sudo su
[sudo] password for ismidz:
root@PC:~# gcc stat.c
stat.c: In function 'main':
stat.c:19:24: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blksize_t {aka long int}' [-Wformat=]
  printf("Block size : %d\n", file.st_blksize);
                        ^~
                        %ld
stat.c:20:30: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blkcnt_t {aka long int}' [-Wformat=]
  printf("Blocks allocated : %d\n", file.st_blocks);
                             ^~
                             %ld
stat.c:21:23: warning: format '%d' expects argument of type 'int', but argument 2 has type '__ino_t {aka long unsigned int}' [-Wformat=]
  printf("Inode no. : %d\n", file.st_ino);
                        ^~
                        %ld
stat.c:24:23: warning: format '%d' expects argument of type 'int', but argument 2 has type '__off_t {aka long int}' [-Wformat=]
  printf("File size : %d bytes\n", file.st_size);
                        ^~
                        %ld
stat.c:25:25: warning: format '%d' expects argument of type 'int', but argument 2 has type '__nlink_t {aka long unsigned int}' [-Wformat=]
  printf("No. of link : %d\n", file.st_nlink);
                        ^~
                        %ld
root@PC:~# gcc stat.c
root@PC:~# ./a.out stat.c
User id : 0
Group id : 4096
Block size : 4096
Blocks allocated : 8
Inode no. : 7484040
Last accessed : Thu Nov 21 14:25:33 2019
Last modified : Thu Nov 21 14:25:28 2019

File size : 1370 bytes
No. of link : 1
Permissions : -rw-r--r--
File type : Regular
root@PC:~#
```

## 5. dirlist.c

```
GNU nano 2.9.3 dirlist.c
#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);
}
```

```
ismidz@PC:~$ sudo su
[sudo] password for ismidz:
root@PC:/home/ismidz# gcc dirlist.c
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char*argv[]){
^~~~~
dirlist.c: In function 'main':
dirlist.c:13:22: error: 'argv' undeclared (first use in this function); did you mean 'agv'?
    if((dname = opendir(argv[1])) == NULL)
                       ^~~~~
dirlist.c:13:22: note: each undeclared identifier is reported only once for each function it appears in
root@PC:/home/ismidz# ./a.out dirlist.c
User id : 0
Group id : 0
Block size : 4096
Blocks allocated : 8
Inode no. : 7484011
Last accessed : Thu Nov 21 14:27:47 2019

Last modified : Tue Nov 19 21:23:10 2019

File size : 354 bytes
No. of link : 1
Permissions :-rw-r--r--
File type : Regular
root@PC:/home/ismidz#
```