

PRAKTIKUM SISTEM OPERASI

MODUL 8

“SYSTEM CALL”



DISUSUN OLEH:

ILHAM RIAN NOVANTO

L200200247

INFORMATIKA

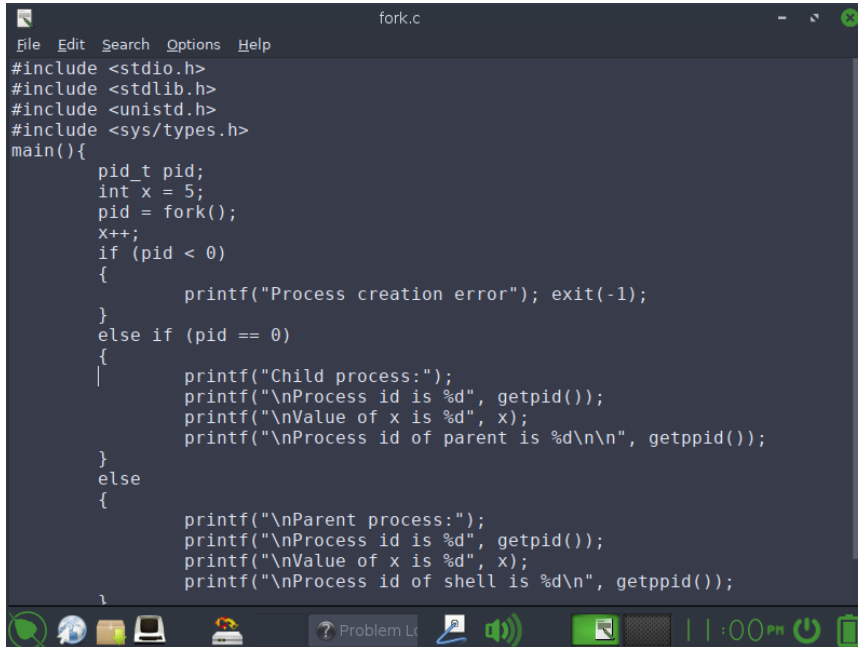
FAKULTAS KOMUNIKASI DAN INFORMATIKA

UNIVERSITAS MUHAMMADIYAH SURAKARTA

2021

FORK.C

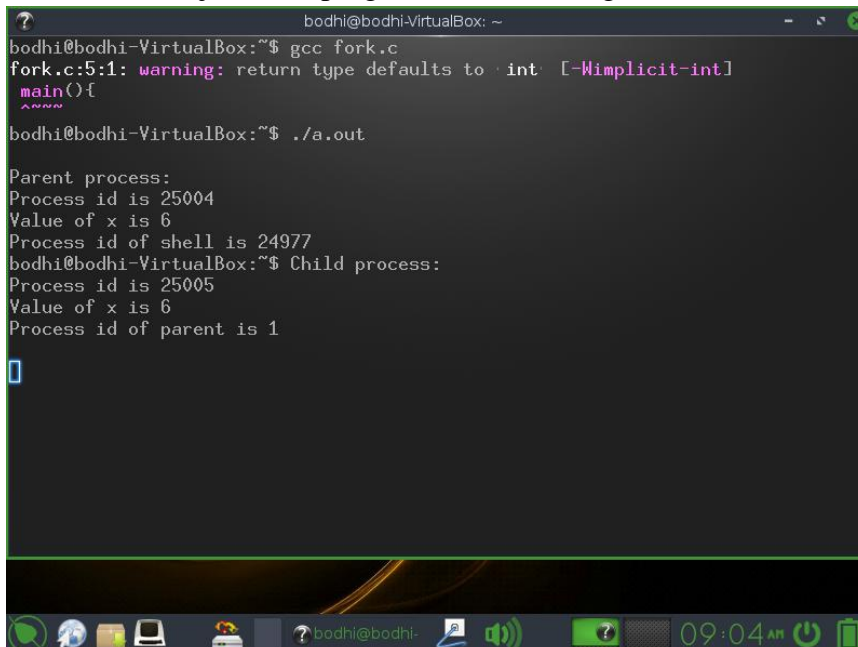
1. Pertama kita membuat program “C” dengan nama “fork” lalu save.

A screenshot of a code editor window titled 'fork.c'. The editor contains the following C code:

```
File Edit Search Options Help
#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("\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 editor has a menu bar with 'File', 'Edit', 'Search', 'Options', and 'Help'. The status bar at the bottom shows 'Problem List' and a timer '1:00'.

2. Lalu compile file tersebut dengan “gcc fork.c”. jika tidak ada kesalahan akan muncul file a.out dan untuk menjalankan program tersebut dengan ./a.out maka hasilnya akan tampak dibawah.

A screenshot of a terminal window titled 'bodhi@bodhi-VirtualBox: ~'. The terminal shows the following commands and output:

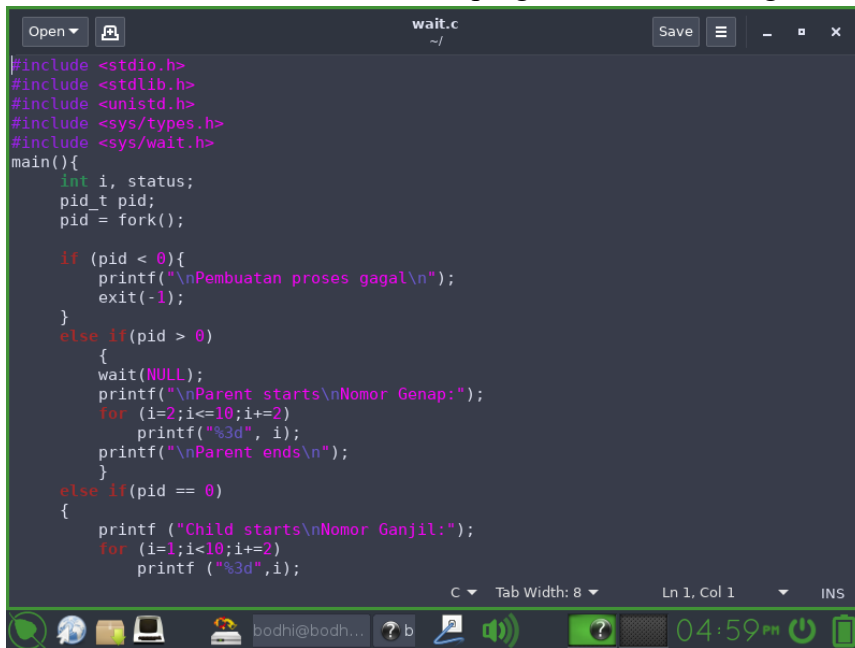
```
bodhi@bodhi-VirtualBox:~$ gcc fork.c
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(){
^~~~~~
bodhi@bodhi-VirtualBox:~$ ./a.out

Parent process:
Process id is 25004
Value of x is 6
Process id of shell is 24977
bodhi@bodhi-VirtualBox:~$ Child process:
Process id is 25005
Value of x is 6
Process id of parent is 1
```

The terminal shows the compilation of 'fork.c' with a warning about the return type of 'main'. It then shows the execution of './a.out', which produces output for both the parent and child processes. The status bar at the bottom shows the time '09:04 AM'.

WAIT.C

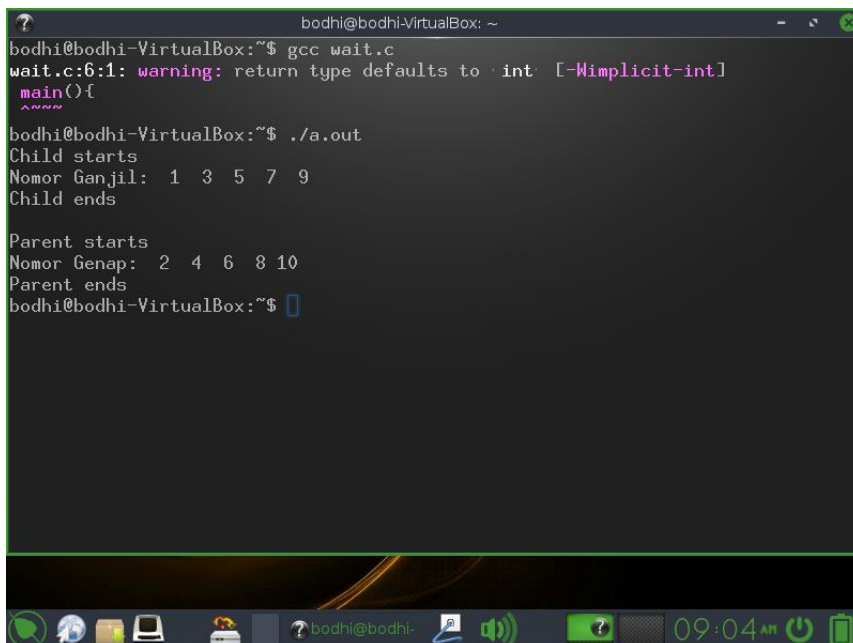
1. Pertama kita membuat program “C” dengan nama “wait” lalu save.



```
#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 starts\nNomor Genap:");
        for (i=2; i<=10; i+=2)
            printf("%3d", i);
        printf("\nParent ends\n");
    }
    else if (pid == 0)
    {
        printf ("Child starts\nNomor Ganjil:");
        for (i=1; i<=10; i+=2)
            printf ("%3d", i);
    }
}
```

2. Lalu compile file tersebut dengan “gcc wait.c”. jika tidak ada kesalahan akan muncul file a.out dan untuk menjalankan program tersebut dengan ./a.out maka hasilnya akan tampak dibawah.

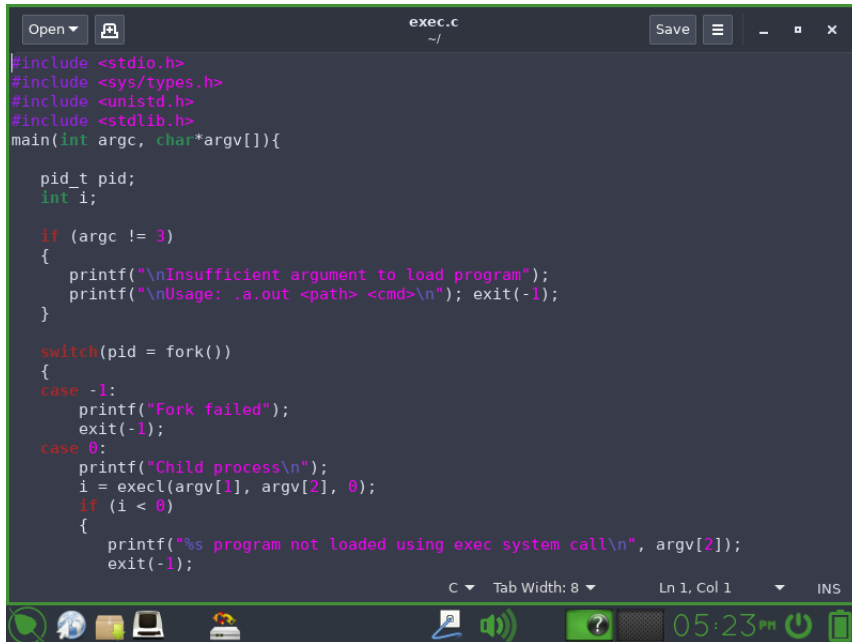


```
bodhi@bodhi-VirtualBox: ~
bodhi@bodhi-VirtualBox:~$ gcc wait.c
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
main()
^~~~~
bodhi@bodhi-VirtualBox:~$ ./a.out
Child starts
Nomor Ganjil:  1  3  5  7  9
Child ends

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

EXEC.C

1. Pertama kita membuat program “C” dengan nama “exec” lalu save.



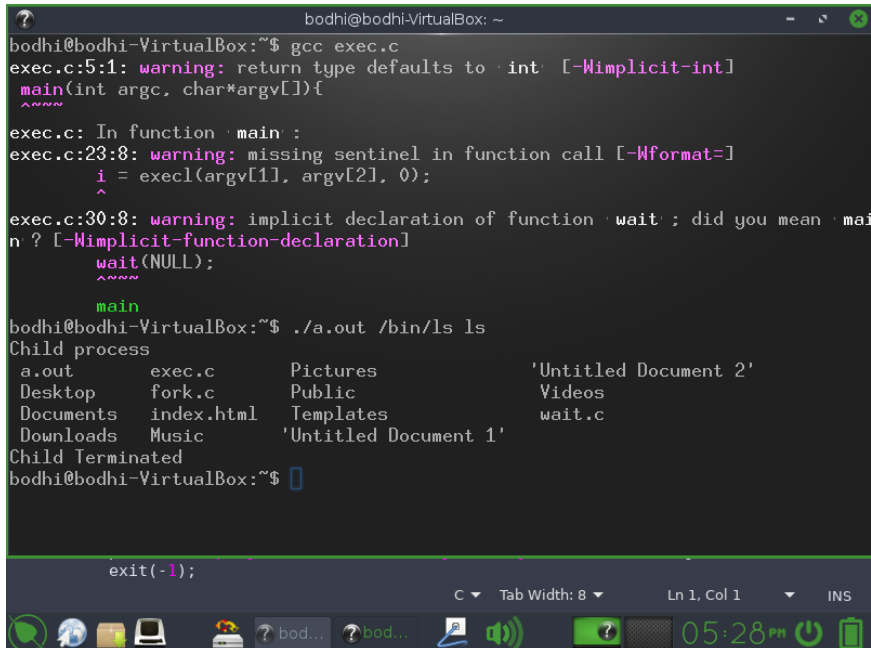
```
Open exec.c Save
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
main(int argc, char*argv[]){

    pid_t pid;
    int i;

    if (argc != 3)
    {
        printf("\nInsufficient argument 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);
        }
    }
```

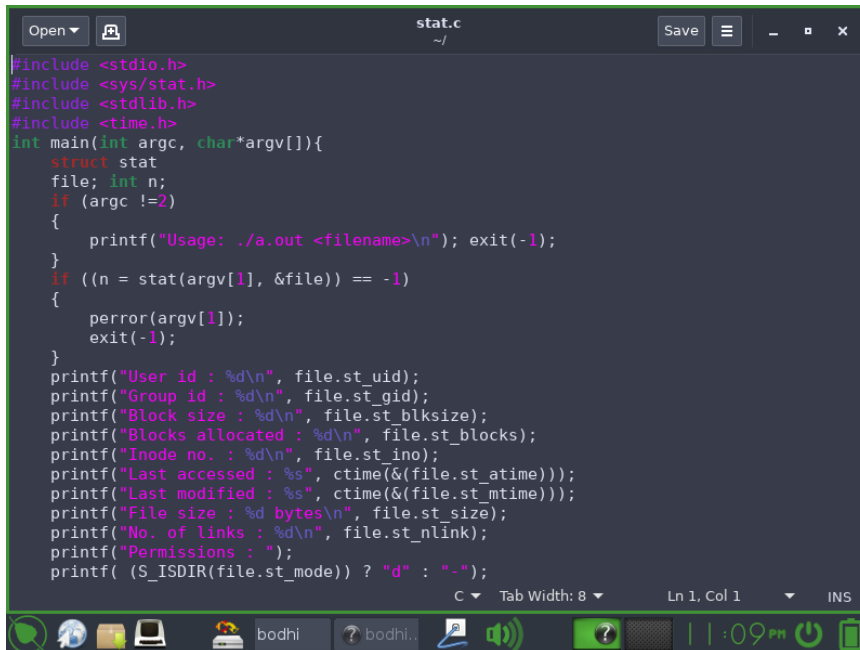
2. Lalu compile file tersebut dengan “gcc exec.c”. jika tidak ada kesalahan akan muncul file a.out dan untuk menjalankan program tersebut dengan ./a.out maka hasilnya akan tampak dibawah.



```
bodhi@bodhi-VirtualBox: ~
bodhi@bodhi-VirtualBox:~$ 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:23:8: warning: missing sentinel in function call [-Wformat=]
    i = execl(argv[1], argv[2], 0);
    ^
exec.c:30:8: warning: implicit declaration of function 'wait'; did you mean 'main'? [-Wimplicit-function-declaration]
    wait(NULL);
    ^~~~~
main
bodhi@bodhi-VirtualBox:~$ ./a.out /bin/ls ls
Child process
a.out      exec.c      Pictures    'Untitled Document 2'
Desktop    fork.c      Public      Videos
Documents  index.html  Templates   wait.c
Downloads  Music       'Untitled Document 1'
Child Terminated
bodhi@bodhi-VirtualBox:~$
```

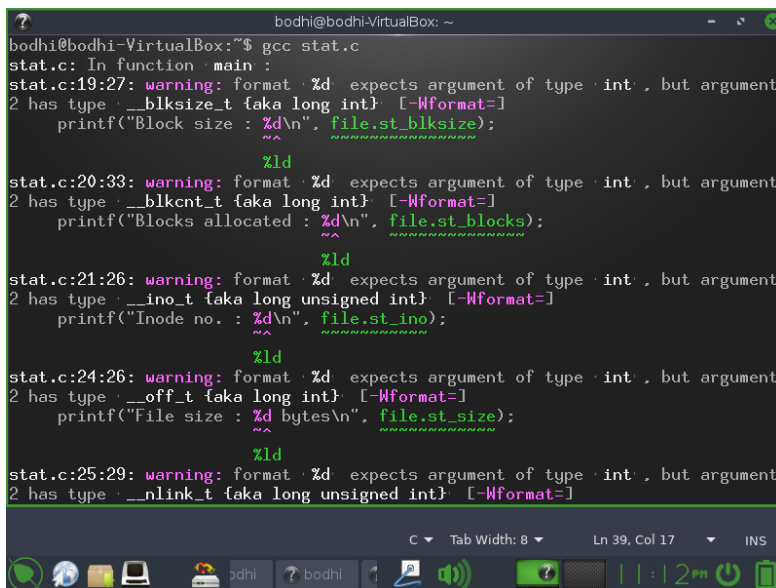
STAT.C

1. Pertama kita membuat program “C” dengan nama “stat” lalu save.



```
Open stat.c 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", ctime(&(file.st_atime)));
    printf("Last modified : %s", 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" : "-");
}
```


2. Lalu compile file tersebut dengan “gcc stat.c”. jika tidak ada kesalahan akan muncul file a.out dan untuk menjalankan program tersebut dengan ./a.out maka hasilnya akan tampak dibawah.



```
bodhi@bodhi-VirtualBox: ~
bodhi@bodhi-VirtualBox:~$ gcc stat.c
stat.c: In function 'main':
stat.c:19:27: 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:20:33: 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:21:26: 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:24:26: 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:25:29: warning: format '%d' expects argument of type 'int', but argument
2 has type '__nlink_t {aka long unsigned int}' [-Wformat=]
```

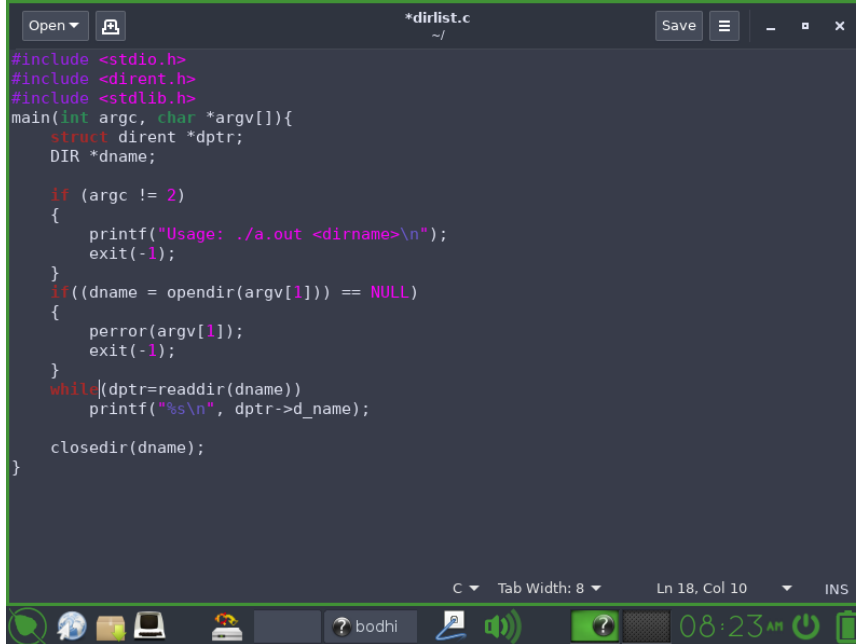
```
bodhi@bodhi-VirtualBox: ~  
  
stat.c:24:26: warning: format '%d' expects argument of type 'int', but argument  
2 has type '__off_t' {aka long int} [-Wformat=]  
    printf("File size : %d bytes\n", file.st_size);  
                        ^~  
                        ~~~~~  
stat.c:25:29: warning: 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);  
                        ^~  
                        ~~~~~  
bodhi@bodhi-VirtualBox:~$ ./a.out fork.c  
User id : 1000  
Group id : 1000  
Block size : 4096  
Blocks allocated : 8  
Inode no. : 266580  
Last accessed : Sun Dec  5 16:15:17 2021  
Last modified : Sun Dec  5 14:53:26 2021  
File size : 682 bytes  
No. of links : 1  
Permissions : -rw-rw-r--  
File type : Regular  
bodhi@bodhi-VirtualBox:~$
```

C Tab Width: 8 Ln 39, Col 17 INS



DIRLIST.C

1. Pertama kita membuat program “C” dengan nama “dirlist” lalu save.

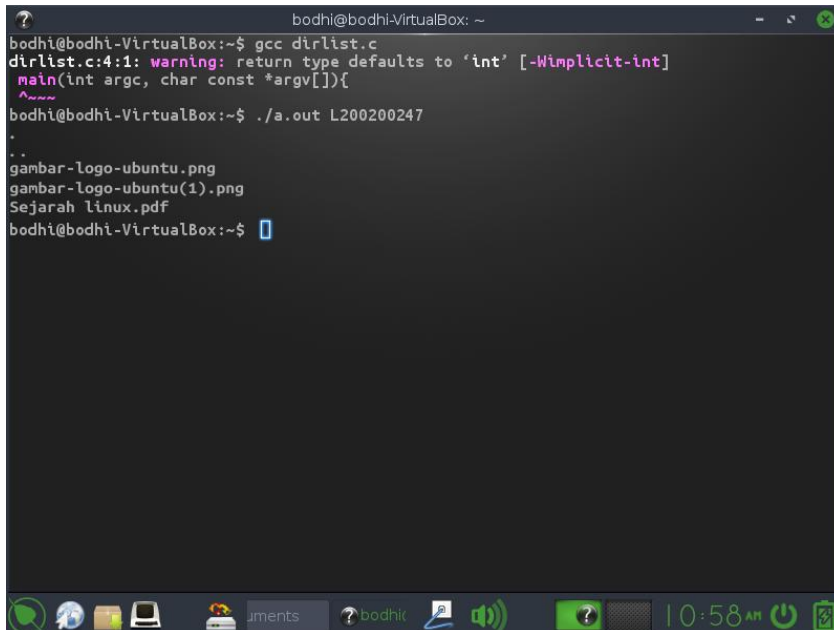
A screenshot of a code editor window titled '*dirlist.c' with a dark theme. The code is a C program that lists the contents of a directory. It includes headers for stdio, dirent, and stdlib. The main function takes two arguments: argc and argv. It checks if argc is not 2, and if so, it prints a usage message and exits. Otherwise, it opens the directory specified in argv[1] using opendir. It then enters a while loop that reads directory entries using readdir and prints the d_name of each entry. Finally, it closes the directory with closedir. The status bar at the bottom shows 'C', 'Tab Width: 8', 'Ln 18, Col 10', and 'INS'.

```
#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);
}
```

2. Lalu compile file tersebut dengan “gcc dirlist.c”. jika tidak ada kesalahan akan muncul file a.out dan untuk menjalankan program tersebut dengan ./a.out maka hasilnya akan tampak dibawah.

A screenshot of a terminal window titled 'bodhi@bodhi-VirtualBox: ~'. It shows the compilation of dirlist.c using gcc, which produces a warning about an implicit int return type. Then, the program is executed with ./a.out L200200247, which lists the contents of the directory L200200247. The output shows files like gambar-logo-ubuntu.png, gambar-logo-ubuntu(1).png, and Sejarah linux.pdf. The prompt returns to the user.

```
bodhi@bodhi-VirtualBox:~$ gcc dirlist.c
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char const *argv[]){
^
bodhi@bodhi-VirtualBox:~$ ./a.out L200200247
..
gambar-logo-ubuntu.png
gambar-logo-ubuntu(1).png
Sejarah linux.pdf
bodhi@bodhi-VirtualBox:~$
```