

**LAPORAN PRAKTIKUM SISTEM OPERASI
MODUL 8
“SYSTEM CALL”**



Oleh:

**NAMA : Daffa Putra Alwansyah
NIM : L200190031
KELAS : B
PRODI : INFORMATIKA**

**Fakultas Komunikasi dan Informatika Universitas
Muhammadiyah Surakarta**

FORK.C

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

```
fork.c
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("Proccess creation error"); exit(-1);
    }
    else if (pid == 0)
    {
        printf("Child proccess:");
        printf("\nProccess id is %d", getpid());
        printf("\nValue of x is %d", x);
        printf("\nProccess id of parent is %d\n\n", getppid());
    }
    else
    {
        printf("\nParent proccess:");
        printf("\nProccess id is %d", getpid());
        printf("\nValue of x is %d", x);
    }
}
```

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.

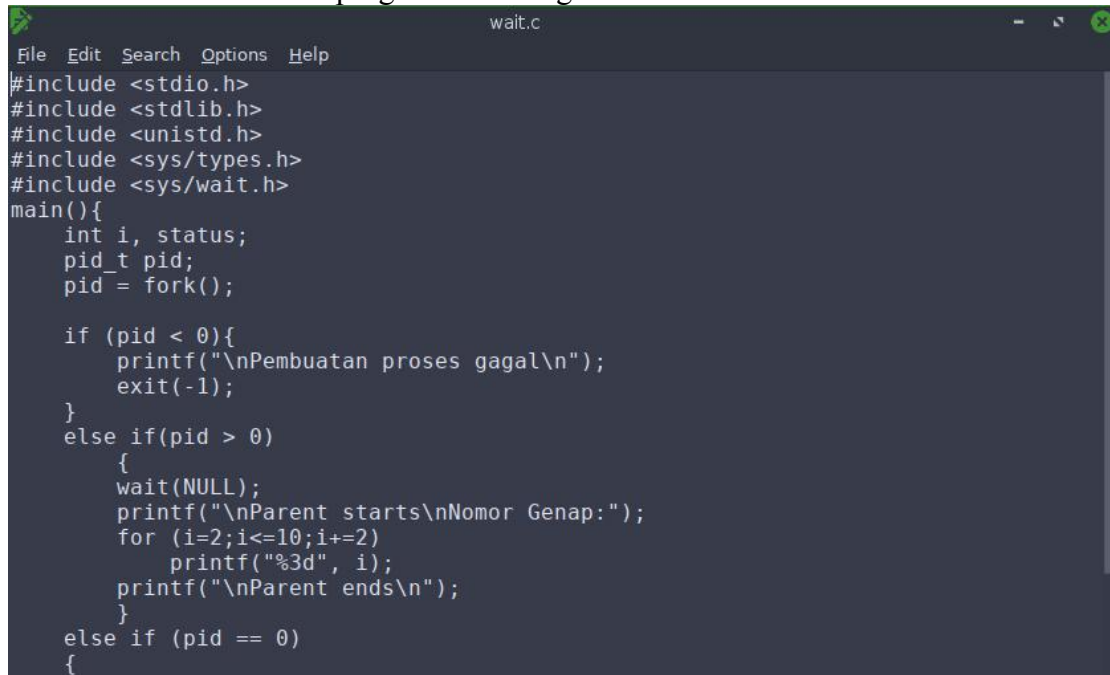
```
bodhi@bodhi-VirtualBox:~$ sudo su
[sudo] password for bodhi:
root@bodhi-VirtualBox:/home/bodhi# cd Documents
root@bodhi-VirtualBox:/home/bodhi/Documents# dir
dirlist.c exec.c fork.c stat.c wait.c
root@bodhi-VirtualBox:/home/bodhi/Documents# gcc fork.c
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main() {
^~~~~
root@bodhi-VirtualBox:/home/bodhi/Documents# ./a.out

Parent proccess:
Proccess id is 31098
Value of x is 6
Proccess id of shell is 31081
root@bodhi-VirtualBox:/home/bodhi/Documents# Child proccess:
Proccess id is 31099
Value of x is 6
Proccess id of parent is 1

root@bodhi-VirtualBox:/home/bodhi/Documents#
```

WAIT.C

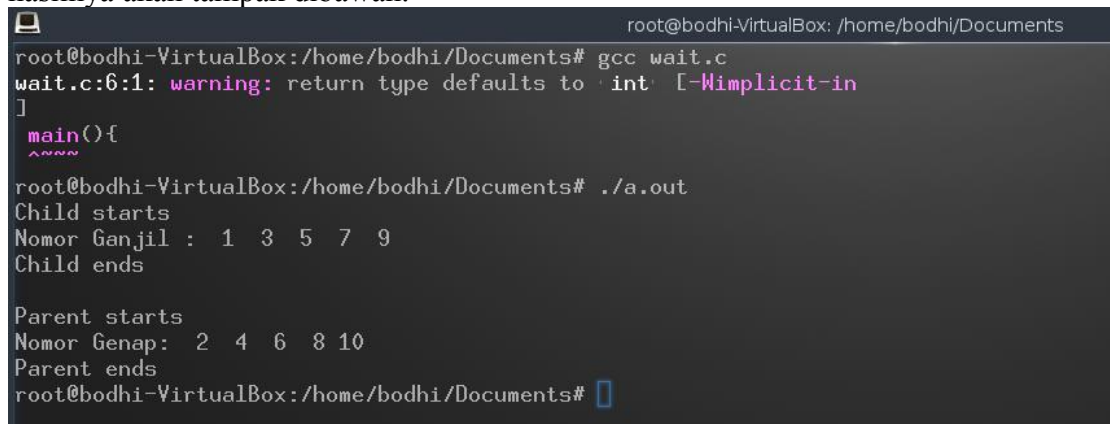
1. Pertama kita membuat program “C” dengan nama “wait” lalu save.



```
File Edit Search Options Help
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("\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)
    {
```

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.



```
root@bodhi-VirtualBox: /home/bodhi/Documents
root@bodhi-VirtualBox:/home/bodhi/Documents# gcc wait.c
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
]
main(){
~~~~~
root@bodhi-VirtualBox:/home/bodhi/Documents# ./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@bodhi-VirtualBox:/home/bodhi/Documents#
```

EXEC.C

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

```
exec.c
File Edit Search Options Help
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
main(int argc, char const *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 proccess\n");
            i = execl(argv[1], argv[2], 0);
            if (i < 0)
            {

```

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.

```
root@bodhi-VirtualBox:/home/bodhi/Documents# gcc exec.c

exec.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char const *argv[])
^~~~~
exec.c: In function 'main':
exec.c:22:13: warning: missing sentinel in function call [-Wformat=]
    i = execl(argv[1], argv[2], 0);
          ^
exec.c:29:13: warning: implicit declaration of function 'wait'; did you mean 'main'? [-Wimplicit-function-declaration]
    wait(NULL);
    ^~~~~
main
root@bodhi-VirtualBox:/home/bodhi/Documents# ./a.out /bin/ls ls
Child proccess
a.out dirlist.c exec.c fork.c stat.c wait.c
Child Terminated
root@bodhi-VirtualBox:/home/bodhi/Documents#
```

STAT.C

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

```
stat.c
File Edit Search Options Help
#include <stdio.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <time.h>
int main(int argc, char const *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 modified : %s\n", ctime(&(file.st_mtime)));
    printf("File size : %d bytes\n", file.st_size);
    printf("No. of links : %d\n", file.st_nlink);
}
```

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.

```
root@bodhi-VirtualBox:/home/bodhi/Documents# gcc stat.c

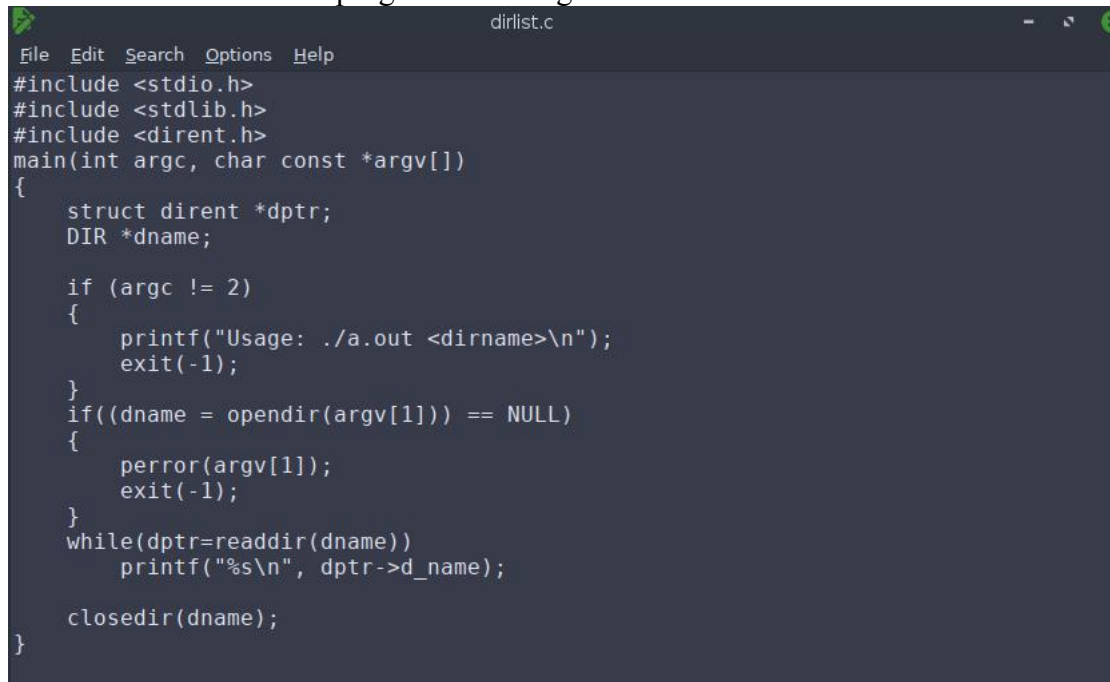
stat.c: In function 'main':
stat.c:19:27: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blksize_t' {aka long int} [-Wformat=]
    printf("Block size : %d\n", file.st_blksize);
                           ^~
stat.c:20:33: warning: format '%d' expects argument of type 'int', but argument 2 has type '__blkcnt_t' {aka long int} [-Wformat=]
    printf("Blocks allocated : %d\n", file.st_blocks);
                               ^~
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);
                           ^~
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);
                           ^~

root@bodhi-VirtualBox:/home/bodhi/Documents# ./a.out fork.c
User id : 1000
Group id : 1000
Block size : 4096
Blocks allocated : 8
Inode no. : 1200352
Last accessed : Fri Dec 11 15:03:03 2020
Last modified : Fri Dec 11 14:56:39 2020

File size : 665 bytes
No. of links : 1
Permissions : -rw-rw-r--
File type : Regular
root@bodhi-VirtualBox:/home/bodhi/Documents#
```

DIRLIST.C

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

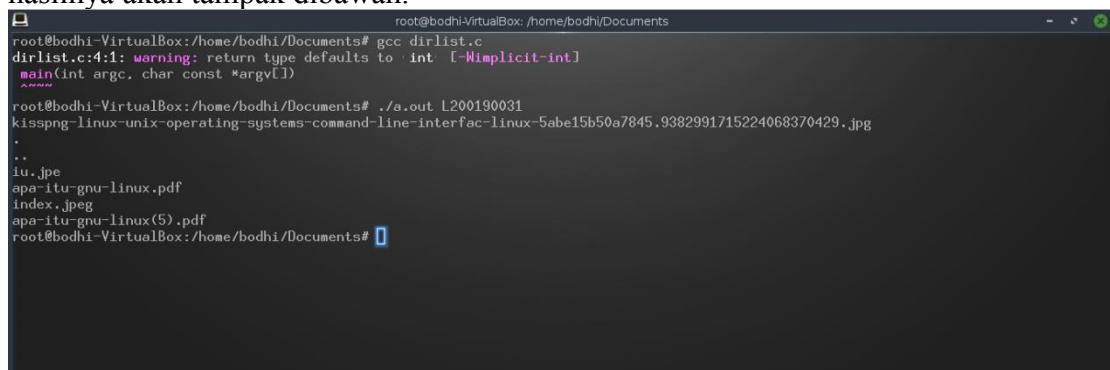
A screenshot of a code editor window titled 'dirlist.c'. The editor has a menu bar with 'File', 'Edit', 'Search', 'Options', and 'Help'. The code is written in C and lists the contents of a directory. It includes headers for stdio, stdlib, and dirent. The main function takes two arguments: argc and argv. It declares a struct dirent pointer and a DIR pointer. It checks if argc is not 2, and if so, prints usage and exits. Otherwise, it opens the directory specified in argv[1] and lists its contents using readdir and printf. Finally, it closes the directory.

```
File Edit Search Options Help
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
main(int argc, char const *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 in a VirtualBox environment. The prompt is 'root@bodhi-VirtualBox: /home/bodhi/Documents'. The user runs 'gcc dirlist.c', which produces a warning about the return type of 'main' and creates 'a.out'. Then, the user runs './a.out', which lists the contents of the current directory. The output shows files like 'kisspng-linux-unix-operating-systems-command-line-interfac-linux-5abe15b50a7845.9382991715224068370429.jpg', 'lu.jpg', 'apa-itu-gnu-linux.pdf', and 'index.jpeg'.

```
root@bodhi-VirtualBox: /home/bodhi/Documents
root@bodhi-VirtualBox: /home/bodhi/Documents# gcc dirlist.c
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char const *argv[])
^~~~~~
root@bodhi-VirtualBox: /home/bodhi/Documents# ./a.out
kisspng-linux-unix-operating-systems-command-line-interfac-linux-5abe15b50a7845.9382991715224068370429.jpg
..
lu.jpg
apa-itu-gnu-linux.pdf
index.jpeg
apa-itu-gnu-linux(5).pdf
root@bodhi-VirtualBox: /home/bodhi/Documents#
```