

Nama : Resya Lusiara	Nilai
Nim : L200210273	
Nama dosen : Heru Setiya Nugraha, ST, M.kom	Tanda tangan
Kelas : Praktikum Sistem Operasi	

MODUL 8

1.

```

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 of parent is %d\n\n", getppid());
20    }
21    else
22    {
23        printf("\nParent process:");
24        printf("\nProcess id is %d", getpid());
25        printf("\nValue of x is %d", x);
26        printf("\nProcess id of parent is %d\n\n", getppid());
27    }
28 }

```

```

resya@resya-VirtualBox: ~
resya@resya-VirtualBox:~$ gcc 'fork.c'
fork.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]
5 | main() {
  | ^~~~
resya@resya-VirtualBox:~$

```

```
resya@resya-VirtualBox:~$ ./a.out
```

```
Parent process:
```

```
Process id is 32862
```

```
Value of x is 6
```

```
Process id of parent is 2351d
```

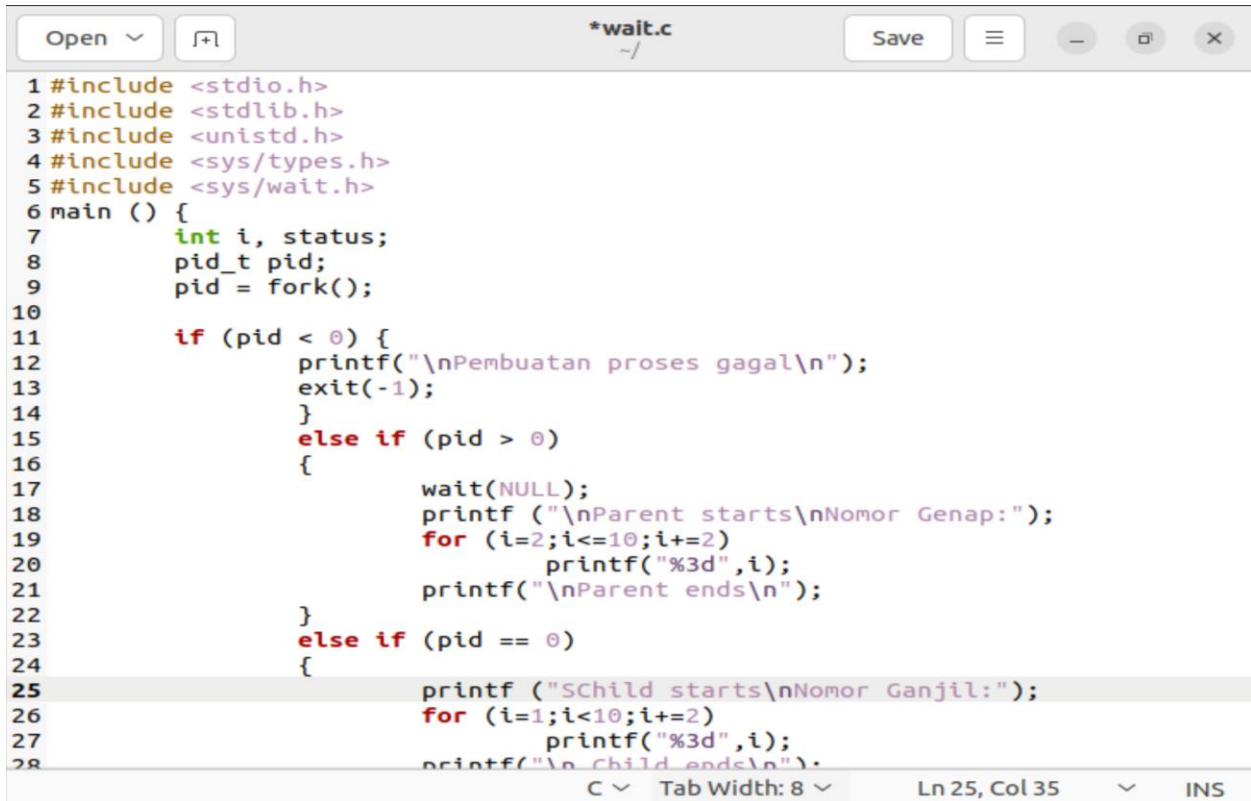
```
resya@resya-VirtualBox:~$ Child process:
```

```
Process id is 32863
```

```
Value of x is 6
```

```
Process id of parent is 1392d
```

2.



```
*wait.c
~/
Save
+
Open
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 }
```

C Tab Width: 8 Ln 25, Col 35 INS

```
resya@resya-VirtualBox: ~  
resya@resya-VirtualBox:~$ gcc 'wait.c'  
wait.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]  
    6 | main () {  
      | ^~~~~~  
resya@resya-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  
resya@resya-VirtualBox:~$
```

3.

```
Open  exec.c  Save  
1 #include <stdio.h>  
2 #include <sys/types.h>  
3 #include <unistd.h>  
4 #include <stdlib.h>  
5 main(int argc, char*argv[]){  
6  
7     pid_t pid;  
8     int i;  
9  
10    if (argc != 3)  
11    {  
12        printf("\nInsufficient arguments to load program");  
13        printf("\nUsage: ./a.out <path> <cmd>\n"); exit(-1);  
14    }  
15  
16    switch(pid = fork())  
17    {  
18    case -1:  
19        printf ("Fork failed");  
20        exit(-1);  
21    case 0:  
22        printf("Child process\n");  
23        i = exec1(argv[1],argv[2],0);  
24        if (i < 0)  
25        {  
26            printf("%s program not loaded using exec system  
call\n", argv[2]);  
27            exit(-1);  
28        }  
29    }  
30}
```

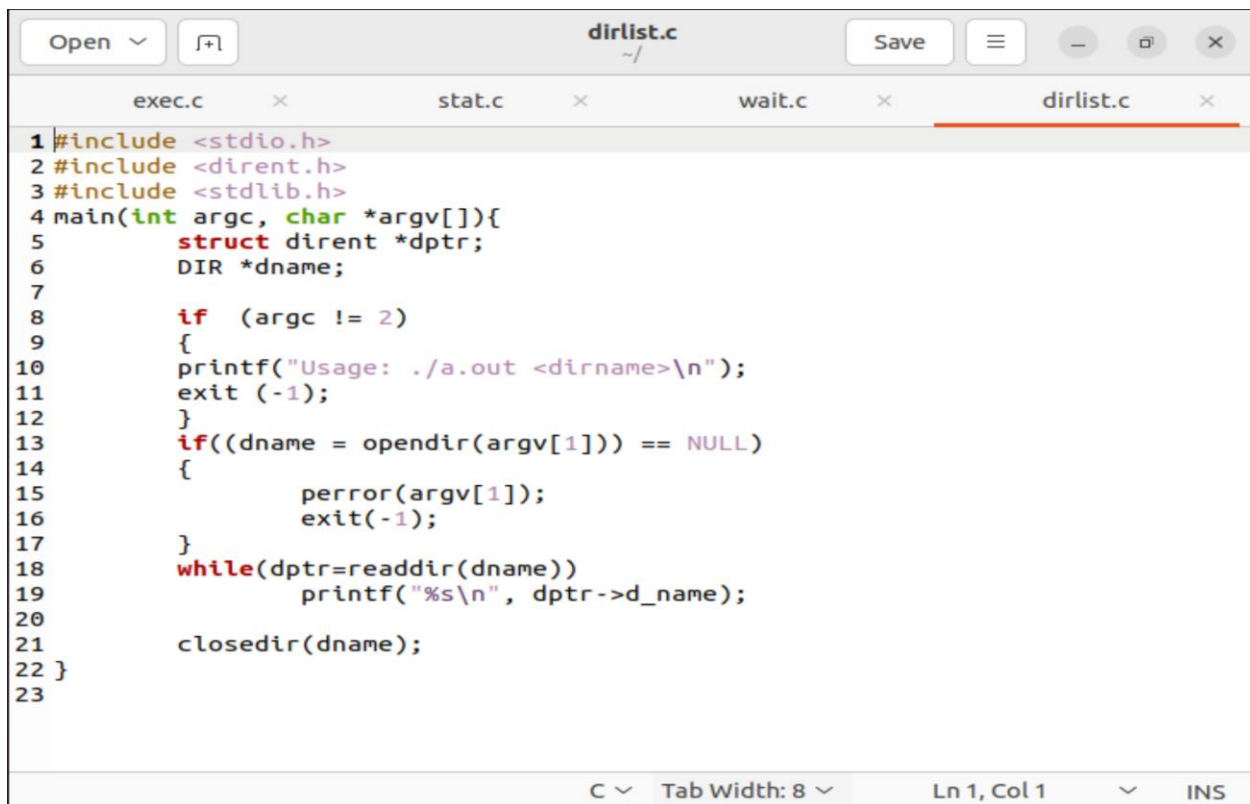
```
resya@resya-VirtualBox: ~  
resya@resya-VirtualBox:~$ gcc exec.c  
exec.c:5:1: warning: return type defaults to 'int' [-Wimplicit-int]  
  5 | main(int argc, char*argv[]){  
    | ^~~~~  
exec.c: In function 'main':  
exec.c:23:17: warning: missing sentinel in function call [-Wformat=]  
  23 |         i = execl(argv[1], argv[2], 0);  
    |         ^  
exec.c:30:17: warning: implicit declaration of function 'wait' [-Wimplicit-func  
tion-declaration]  
  30 |         wait(NULL);  
    |         ^~~~~  
resya@resya-VirtualBox:~$ ./a.out /bin/ls ls  
Child process  
a.out  dirlist.c  Downloads  fork.c  Pictures  snap  Templates  wait.c  
Desktop  Documents  exec.c    Music   Public   stat.c  Videos  
Child Terminal
```

4.

```
stat.c  
~/  
Save  
exec.c  stat.c  
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("Block allocated : %d\n", file.st_blocks);  
21 printf("Inode no. : %d\n", file.st_ino);  
22 printf("Last accesse : %s", ctime(&(file.st_atime)));  
23 printf("Last modified : %s", ctime(&(file.st_mtime)));  
24 printf("File size : %d bytes\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");  
C Tab Width: 8 Ln 1, Col 1 INS
```

```
resya@resya-VirtualBox:~$ ./a.out stat.c
User id : 1000
Group id : 1000
Block size : 4096
Block allocated : 8
Inode no. : 1574656
Last accesse : Sun Dec 11 08:07:09 2022
Last modified : Sat Dec 10 14:41:57 2022
File size : 1371 bytes
No. of links : 1
Permissions : -rw-rw-r--
File type : Regular
resya@resya-VirtualBox:~$
```

5.



```
1 #include <stdio.h>
2 #include <dirent.h>
3 #include <stdlib.h>
4 main(int argc, char *argv[]){
5     struct dirent *dptr;
6     DIR *dname;
7
8     if (argc != 2)
9     {
10         printf("Usage: ./a.out <dirname>\n");
11         exit (-1);
12     }
13     if((dname = opendir(argv[1])) == NULL)
14     {
15         perror(argv[1]);
16         exit(-1);
17     }
18     while(dptr=readdir(dname))
19         printf("%s\n", dptr->d_name);
20
21     closedir(dname);
22 }
23
```

```
resya@resya-VirtualBox: ~  
resya@resya-VirtualBox:~$ gcc dirlist.c  
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]  
 4 | main(int argc, char *argv[]){  
    | ^~~~~  
resya@resya-VirtualBox:~$ ./a.out dirlist.c  
dirlist.c: Not a directory  
resya@resya-VirtualBox:~$ gcc dirlist.c  
dirlist.c:4:1: warning: return type defaults to 'int' [-Wimplicit-int]  
 4 | main(int argc, char *argv[]){  
    | ^~~~~  
resya@resya-VirtualBox:~$ ./a.out Praktikum  
..  
.  
dirlist.c  
resya@resya-VirtualBox:~$
```