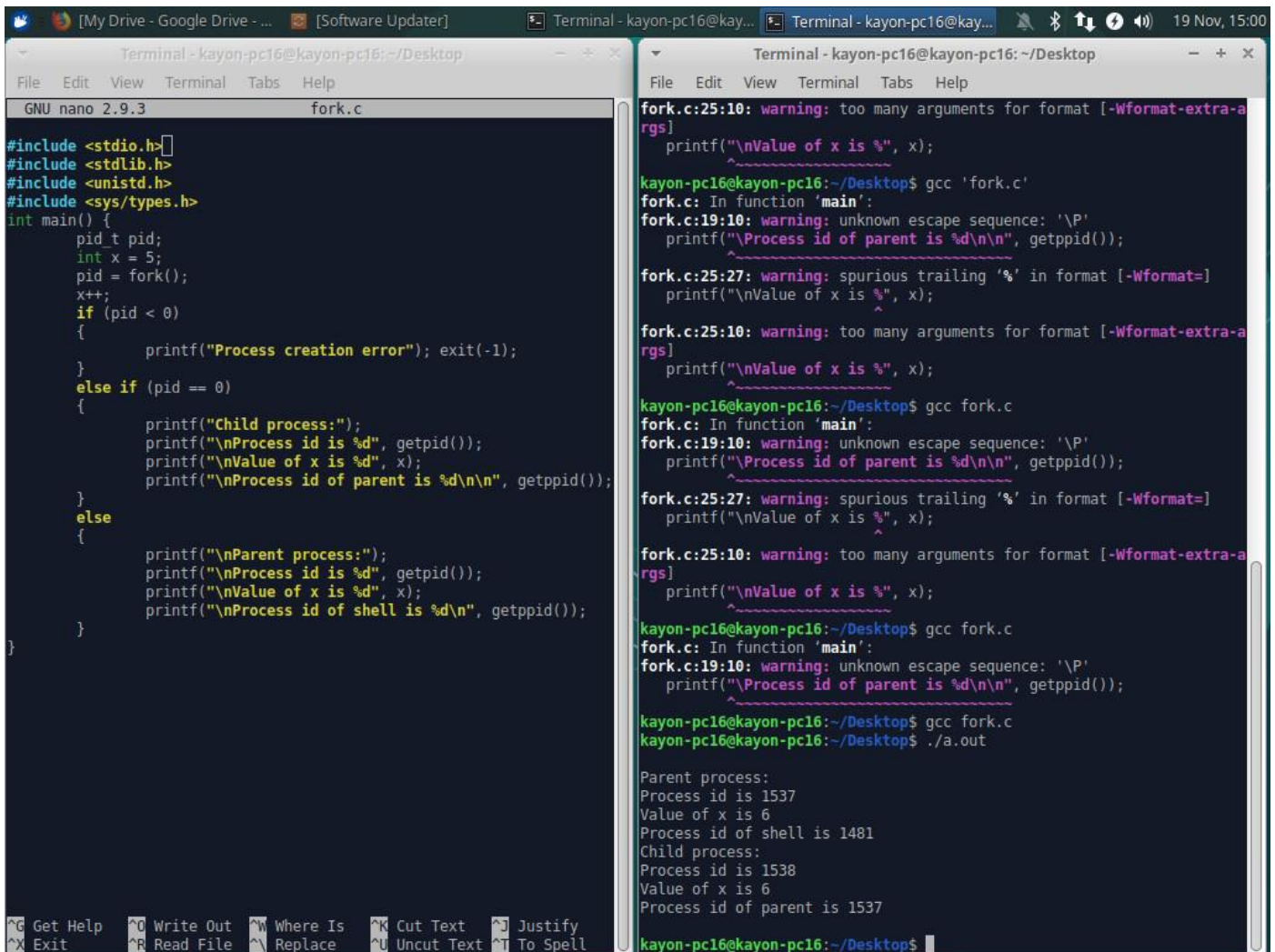


Nama : M. Jainal A
NIM : L200180125
Kelas : C

MODUL 8

1. Membuat child "Child Process" dengan menggunakan System Call "Fork"

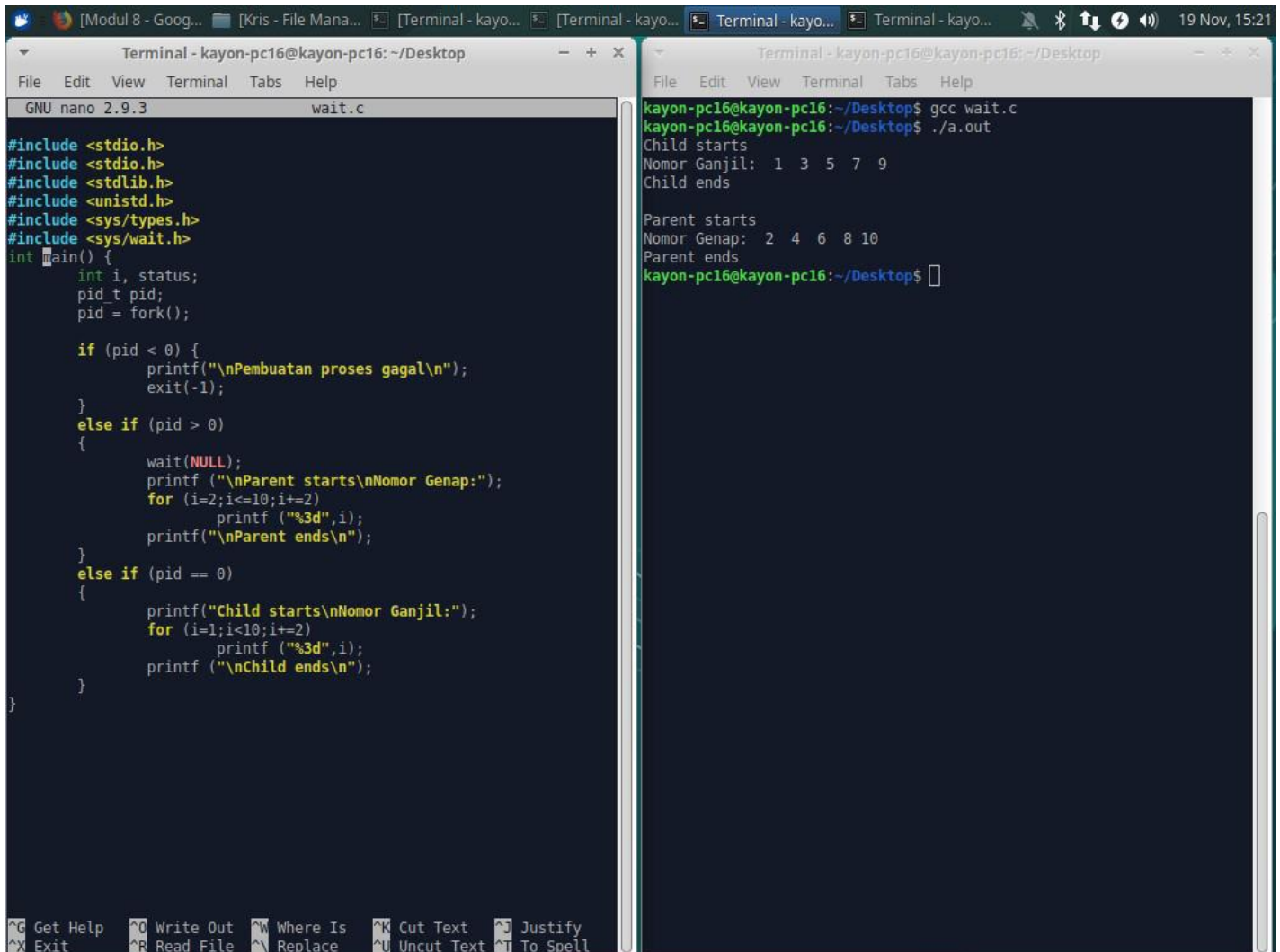


```
GNU nano 2.9.3 fork.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
int main() {
    pid_t pid;
    int x = 5;
    pid = fork();
    x++;
    if (pid < 0)
    {
        printf("Process creation error"); exit(-1);
    }
    else if (pid == 0)
    {
        printf("Child process:");
        printf("\nProcess id is %d", getpid());
        printf("\nValue of x is %d", x);
        printf("\nProcess id of parent is %d\n", 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());
    }
}
```

```
fork.c:25:10: warning: too many arguments for format [-Wformat-extra-args]
    printf("\nValue of x is %", x);
    ^
kayon-pc16@kayon-pc16:~/Desktop$ gcc 'fork.c'
fork.c: In function 'main':
fork.c:19:10: warning: unknown escape sequence: '\P'
    printf("\Process id of parent is %d\n\n", getppid());
    ^
fork.c:25:27: warning: spurious trailing '%' in format [-Wformat=]
    printf("\nValue of x is %", x);
    ^
fork.c:25:10: warning: too many arguments for format [-Wformat-extra-args]
    printf("\nValue of x is %", x);
    ^
kayon-pc16@kayon-pc16:~/Desktop$ gcc fork.c
fork.c: In function 'main':
fork.c:19:10: warning: unknown escape sequence: '\P'
    printf("\Process id of parent is %d\n\n", getppid());
    ^
fork.c:25:27: warning: spurious trailing '%' in format [-Wformat=]
    printf("\nValue of x is %", x);
    ^
fork.c:25:10: warning: too many arguments for format [-Wformat-extra-args]
    printf("\nValue of x is %", x);
    ^
kayon-pc16@kayon-pc16:~/Desktop$ gcc fork.c
kayon-pc16@kayon-pc16:~/Desktop$ ./a.out

Parent process:
Process id is 1537
Value of x is 6
Process id of shell is 1481
Child process:
Process id is 1538
Value of x is 6
Process id of parent is 1537
kayon-pc16@kayon-pc16:~/Desktop$
```

2. Menghentikan sementara (block) proses parent dengan proses child selesai, menggunakan system call "wait"



```
Terminal - kayon-pc16@kayon-pc16: ~/Desktop
GNU nano 2.9.3      wait.c

#include <stdio.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
int 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);
        printf ("\nChild ends\n");
    }
}

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify
^X Exit      ^R Read File  ^N Replace   ^U Uncut Text ^I To Spell

Terminal - kayon-pc16@kayon-pc16:~/Desktop
kayon-pc16@kayon-pc16:~/Desktop$ gcc wait.c
kayon-pc16@kayon-pc16:~/Desktop$ ./a.out
Child starts
Nomor Ganjil:  1  3  5  7  9
Child ends

Parent starts
Nomor Genap:   2  4  6  8 10
Parent ends
kayon-pc16@kayon-pc16:~/Desktop$
```

3. Loading Program yang dapat dieksekusi dalam sebuah “Child” Proses menggunakan perintah system call “exec”

```
Terminal - kayon-pc25@kayon-pc25: ~
File Edit View Terminal Tabs Help

root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~  root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~  root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~

kayon-pc25@kayon-pc25:~$ 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
kayon-pc25@kayon-pc25:~$ ./a.out /bin/ls ls
Child process
a.out Desktop Documents Downloads exec.c fork.c info1200180110.sh.save info1200180110.sh.save.1 Music Pictures Public Templates Videos wait.c
Child Terminated
kayon-pc25@kayon-pc25:~$ 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:27:3: warning: implicit declaration of function 'wait'; did you mean 'main'? [-Wimplicit-function-declaration]
    wait(NULL);
    ^
main
kayon-pc25@kayon-pc25:~$ gcc exec.c
exec.c:5:10: fatal error: sys/wait.h: No such file or directory
#include <sys/wait.h>
        ^
compilation terminated.
kayon-pc25@kayon-pc25:~$ gcc exec.c
exec.c:6:1: warning: return type defaults to 'int' [-Wimplicit-int]
main(int argc, char*argv[]) {
^
kayon-pc25@kayon-pc25:~$ gcc exec.c
kayon-pc25@kayon-pc25:~$ ./a.out /bin/ls ls
Child process
a.out Desktop Documents Downloads exec.c fork.c info1200180110.sh.save info1200180110.sh.save.1 Music Pictures Public Templates Videos wait.c
Child Terminated
kayon-pc25@kayon-pc25:~$
```

```
Terminal - root@kayon-pc25: ~/home/kayon-pc25
File Edit View Terminal Tabs Help

root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~  root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~  root@kayon-pc25: ~/home/kay...  kayon-pc25@kayon-pc25: ~

GNU nano 2.9.3
exec.c
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
#include <sys/wait.h>
int 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], NULL);
            if (i < 0)
            {
                printf("No program not loaded using exec system call\n", argv[2]);
                exit(-1);
            }
        default:
            wait(NULL);
            printf("Child Terminated\n");
            exit(-1);
    }
}
```

4. Menambahkan status file menggunakan perintha system call “stat”

```
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 : %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("Permission : ");
}
```

```
GNU nano 2.9.3 stat.c
    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("Permission : ");
printf(" (%s_ISDIR(file.st_mode)) ? \"d\" : \"-\"");
printf(" (%s_IRUSR & file.st_mode & S_IRUSR) ? \"r\" : \"-\"");
printf(" (%s_IWUSR & file.st_mode & S_IWUSR) ? \"w\" : \"-\"");
printf(" (%s_IXUSR & file.st_mode & S_IXUSR) ? \"x\" : \"-\"");
printf(" (%s_IRGRP & file.st_mode & S_IRGRP) ? \"r\" : \"-\"");
printf(" (%s_IWGRP & file.st_mode & S_IWGRP) ? \"w\" : \"-\"");
printf(" (%s_IXGRP & file.st_mode & S_IXGRP) ? \"x\" : \"-\"");
printf(" (%s_IROTH & file.st_mode & S_IROTH) ? \"r\" : \"-\"");
printf(" (%s_IWOTH & file.st_mode & S_IWOTH) ? \"w\" : \"-\"");
printf(" (%s_IXOTH & file.st_mode & S_IXOTH) ? \"x\" : \"-\"");
printf("\n");
if(file.st_mode & S_IFREG)
    printf("File type : Regular\n");
```

```
GNU nano 2.9.3 stat.c
    printf(" (%s_ISDIR(file.st_mode)) ? \"d\" : \"-\"");
    printf(" (%s_IRUSR & file.st_mode & S_IRUSR) ? \"r\" : \"-\"");
    printf(" (%s_IWUSR & file.st_mode & S_IWUSR) ? \"w\" : \"-\"");
    printf(" (%s_IXUSR & file.st_mode & S_IXUSR) ? \"x\" : \"-\"");
    printf(" (%s_IRGRP & file.st_mode & S_IRGRP) ? \"r\" : \"-\"");
    printf(" (%s_IWGRP & file.st_mode & S_IWGRP) ? \"w\" : \"-\"");
    printf(" (%s_IXGRP & file.st_mode & S_IXGRP) ? \"x\" : \"-\"");
    printf(" (%s_IROTH & file.st_mode & S_IROTH) ? \"r\" : \"-\"");
    printf(" (%s_IWOTH & file.st_mode & S_IWOTH) ? \"w\" : \"-\"");
    printf(" (%s_IXOTH & 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");
```

```
File type : Regular
ftrlag@ftrlag-VirtualBox:~$ gcc stat.c
ftrlag@ftrlag-VirtualBox:~$ ./a.out
Usage: ./a.out <filename>
ftrlag@ftrlag-VirtualBox:~$ gcc stat.c
ftrlag@ftrlag-VirtualBox:~$ ./a.out stat.c
User id : 0
Group id : 0
Block size : 4096
Blocks allocated : 8
Inode no. : 555318
Last accessed : Thu Nov 21 21:09:24 2019
Last modified : Thu Nov 21 21:09:16 2019
File size : 1375 bytes
No. of links : 1
Permission : -rw-r--r--
File type : Regular
ftrlag@ftrlag-VirtualBox:~$ gcc stat.c
ftrlag@ftrlag-VirtualBox:~$ ./a.out stat.c
User id : 0
Group id : 0
Block size : 4096
Blocks allocated : 8
Inode no. : 555318
Last accessed : Thu Nov 21 21:09:24 2019
Last modified : Thu Nov 21 21:09:16 2019
File size : 1375 bytes
No. of links : 1
Permission : -rw-r--r--
File type : Regular
ftrlag@ftrlag-VirtualBox:~$
```

5. Menampilkan isi direktori menggunakan perintah system call "readdir"

```
GNU nano 2.9.3                                dirlist.c

#include <stdio.h>
#include <dirent.h>
#include <stdlib.h>
int 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);
}

Wrote 22 lines
Get Help  Write Out  Where Is  Cut Text  Justify  Cur Pos  M-U Undo  M-A Mark Text  M-J To Bracket
Exit      Read File  Replace  Uncut Text  To Spell  Go To Line  M-E Redo  M-G Copy Text  M-W WhereIs Next
Activate Windows
Go to a setting in Windows.
```

```
fitria@fitria-VirtualBox:~$ gcc dirlist.c
fitria@fitria-VirtualBox:~$ ./a.out dirlist.c
User id : 0
Group id : 0
Block size : 4096
Blocks allocated : 8
Inode no. : 530210
Last accessed : Thu Nov 21 21:17:00 2019
Last modified : Thu Nov 21 21:16:54 2019
File size : 360 bytes
No. of links : 1
Permission : -rw-r--r--
File type : Regular
fitria@fitria-VirtualBox:~$ ./a.out Downloads
.
..
google-chrome-stable_current_amd64.deb
fitria@fitria-VirtualBox:~$ ./a.out Pictures
Screenshot from 2019-11-09 14-45-38.png
.
..
Screenshot from 2019-11-09 14-48-13.png
Screenshot from 2019-11-09 14-48-33.png
fitria@fitria-VirtualBox:~$
```