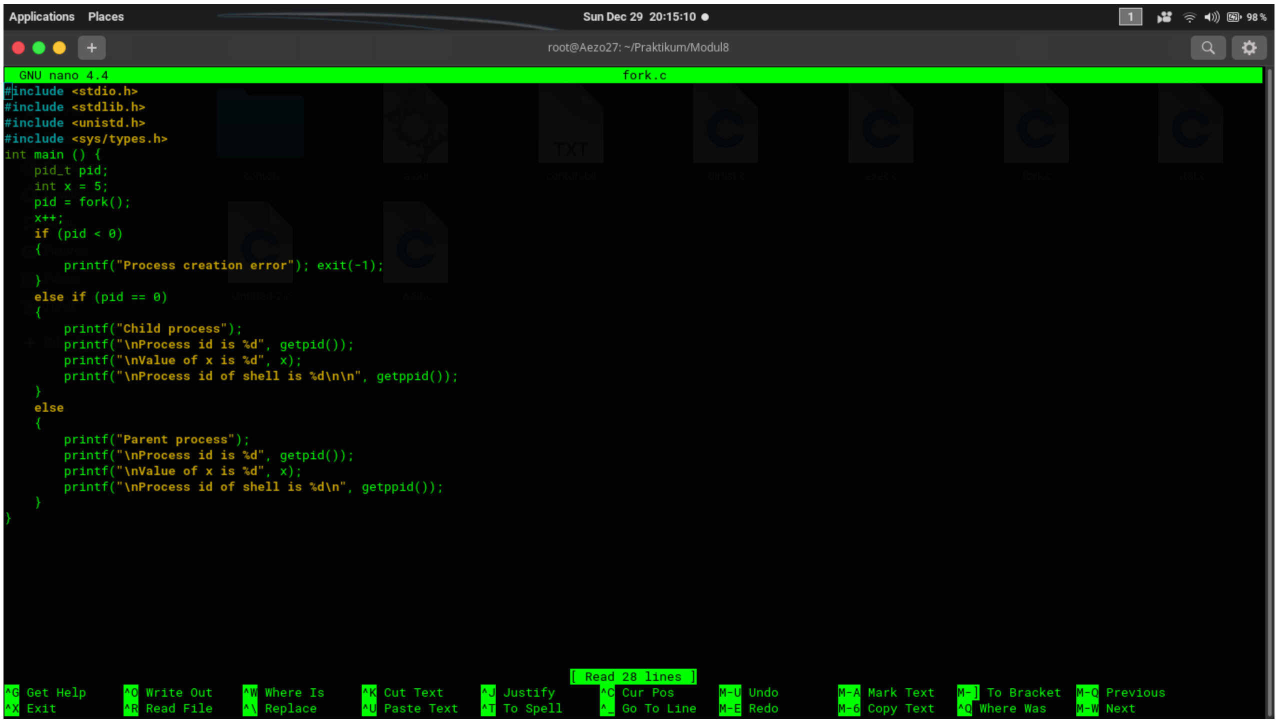
Nama : Diah Fitri Ramadhani

NIM : L200180106

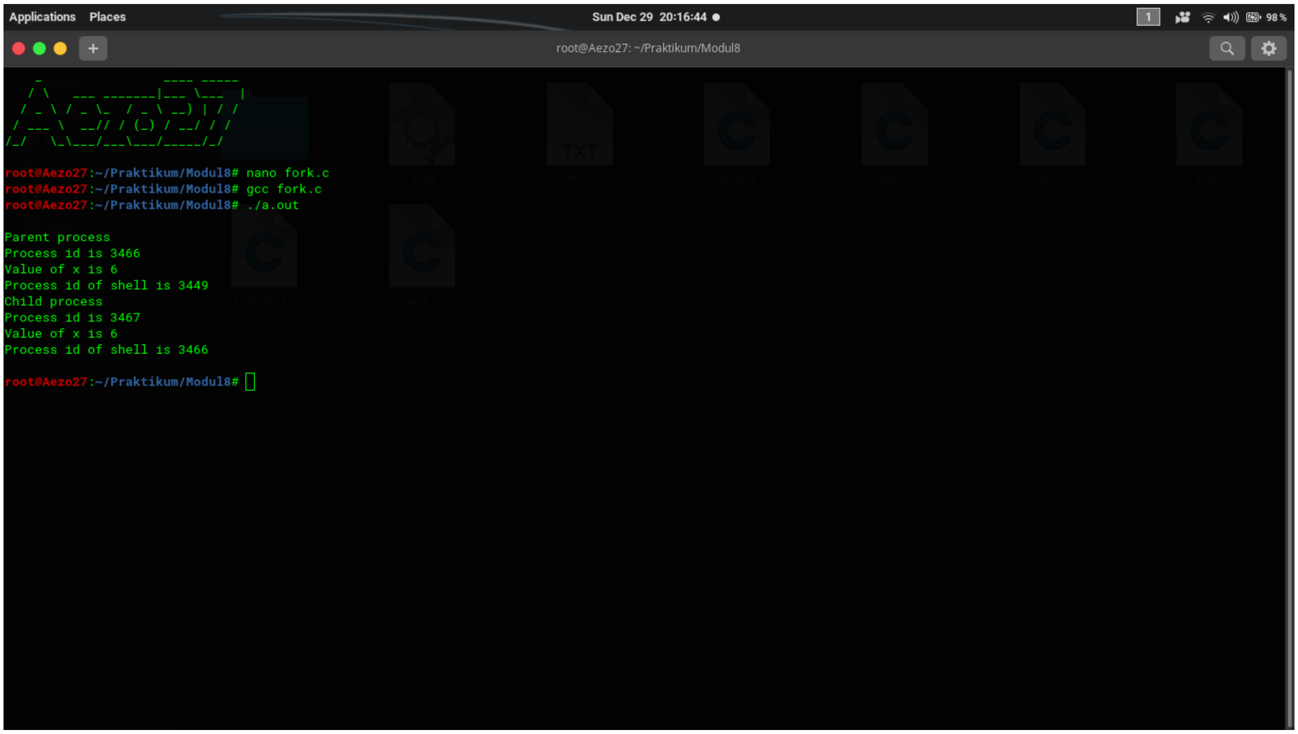
MODUL 8 (KEGIATAN 1 – 3)

Fork.c



```
  GNU nano 4.4                                                                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 shell is %d\n\n", getppid());
    }
    else
    {
        printf("Parent process");
        printf("\nProcess id is %d", getpid());
        printf("\nValue of x is %d", x);
        printf("\nProcess id of shell is %d\n", getppid());
    }
}

                                          [ Read 28 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text  M-] To Bracket M-Q Previous
^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text  ^Q Where Was   M-W Next
```



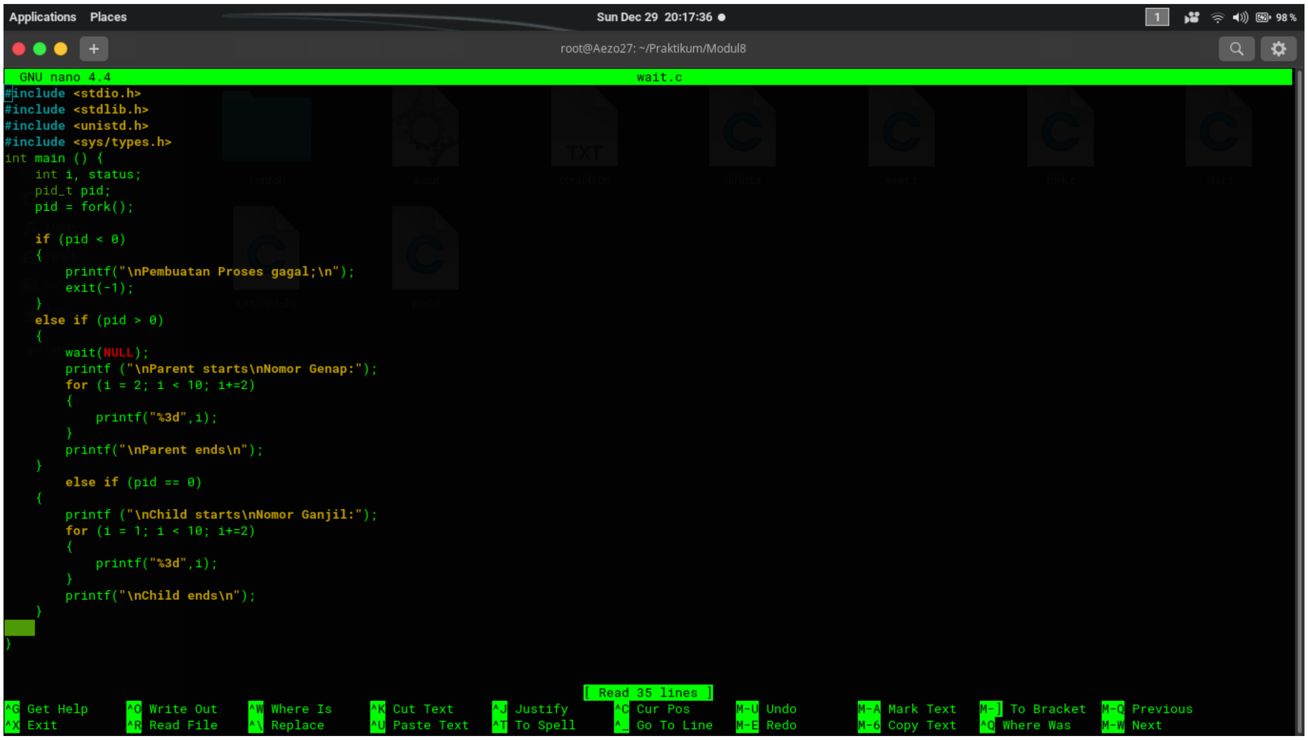
```
    _               ____ _____
   / \   ___ _______|___ \___  |
  / _ \ / _ \_  / _ \ __) | / /
 / ___ \  __// / (_) / __/ / /
/_/   \_\___/___\___/_____/_/

root@Aezo27:~/Praktikum/Modul8# nano fork.c
root@Aezo27:~/Praktikum/Modul8# gcc fork.c
root@Aezo27:~/Praktikum/Modul8# ./a.out

Parent process
Process id is 3466
Value of x is 6
Process id of shell is 3449
Child process
Process id is 3467
Value of x is 6
Process id of shell is 3466

root@Aezo27:~/Praktikum/Modul8# 
```

Wait.c

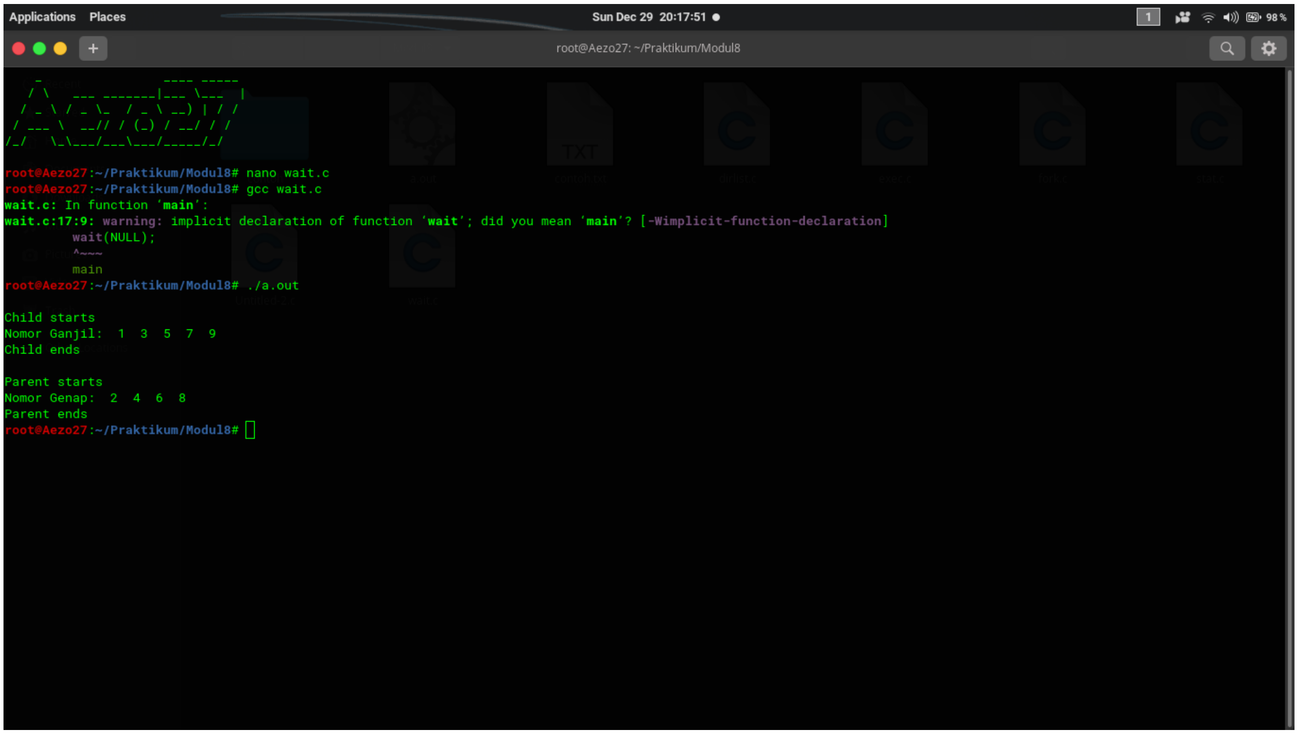


```
  GNU nano 4.4                                wait.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
int main () {
    int i, status;
    pid_t pid;
    pid = fork();

    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 ("\nChild starts\nNomor Ganjil:");
        for (i = 1; i < 10; i+=2)
        {
            printf("%3d",i);
        }
        printf("\nChild ends\n");
    }

}

                                    [ Read 35 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text  M-] To Bracket M-Q Previous
^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text  ^Q Where Was   M-W Next
```



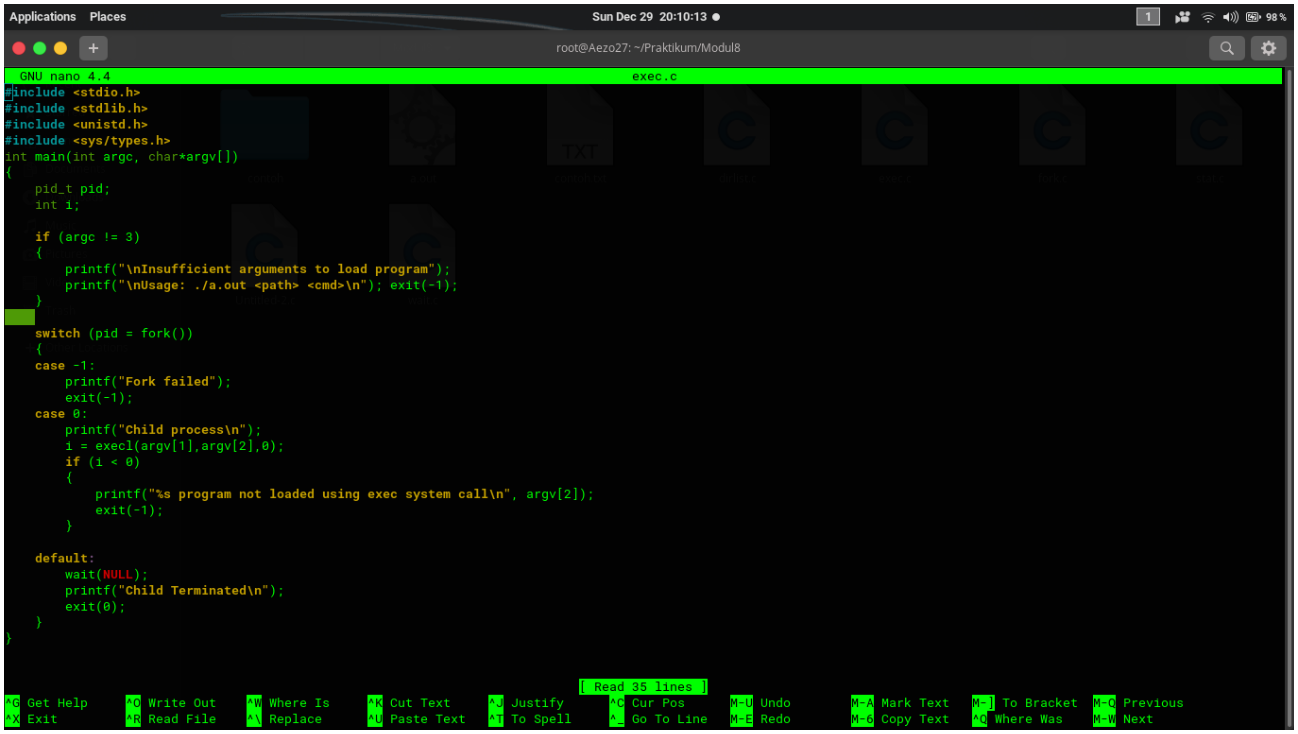
```
    _                ____ _____
   / \   ___ _______|___ \___  |
  / _ \ / _ \_  / _ \ __) | / /
 / ___ \  __// / (_) / __/ / /
/_/   \_\___/___\___/_____/_/

root@Aezo27:~/Praktikum/Modul8# nano wait.c
root@Aezo27:~/Praktikum/Modul8# gcc wait.c
wait.c: In function ‘main’:
wait.c:17:9: warning: implicit declaration of function ‘wait’; did you mean ‘main’? [-Wimplicit-function-declaration]
         wait(NULL);
         ^~~~
         main
root@Aezo27:~/Praktikum/Modul8# ./a.out

Child starts
Nomor Ganjil:  1  3  5  7  9
Child ends

Parent starts
Nomor Genap:  2  4  6  8
Parent ends
root@Aezo27:~/Praktikum/Modul8# 
```

Exec.c



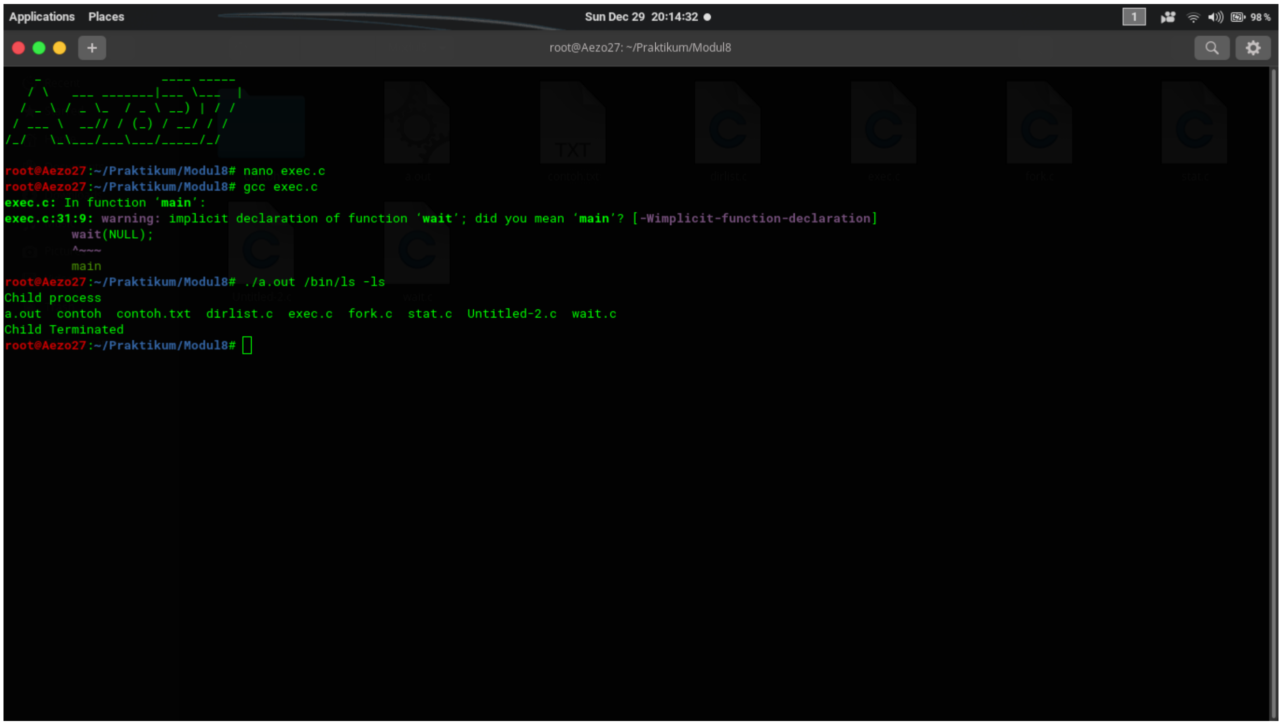
```
  GNU nano 4.4                                          exec.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
int main(int argc, char*argv[])
{
    pid_t pid;
    int i;

    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],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);
    }
}

                                            [ Read 35 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text  M-] To Bracket M-Q Previous
^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text  ^Q Where Was   M-W Next
```



```
    _                ____ _____
   / \   ___ _______|___ \___  |
  / _ \ / _ \_  / _ \ __) | / /
 / ___ \  __// / (_) / __/ / /
/_/   \_\___/___\___/_____/_/

root@Aezo27:~/Praktikum/Modul8# nano exec.c
root@Aezo27:~/Praktikum/Modul8# gcc exec.c
exec.c: In function ‘main’:
exec.c:31:9: warning: implicit declaration of function ‘wait’; did you mean ‘main’? [-Wimplicit-function-declaration]
         wait(NULL);
         ^~~~
         main
root@Aezo27:~/Praktikum/Modul8# ./a.out /bin/ls -ls
Child process
a.out  contoh  contoh.txt  dirlist.c  exec.c  fork.c  stat.c  Untitled-2.c  wait.c
Child Terminated
root@Aezo27:~/Praktikum/Modul8# 
```