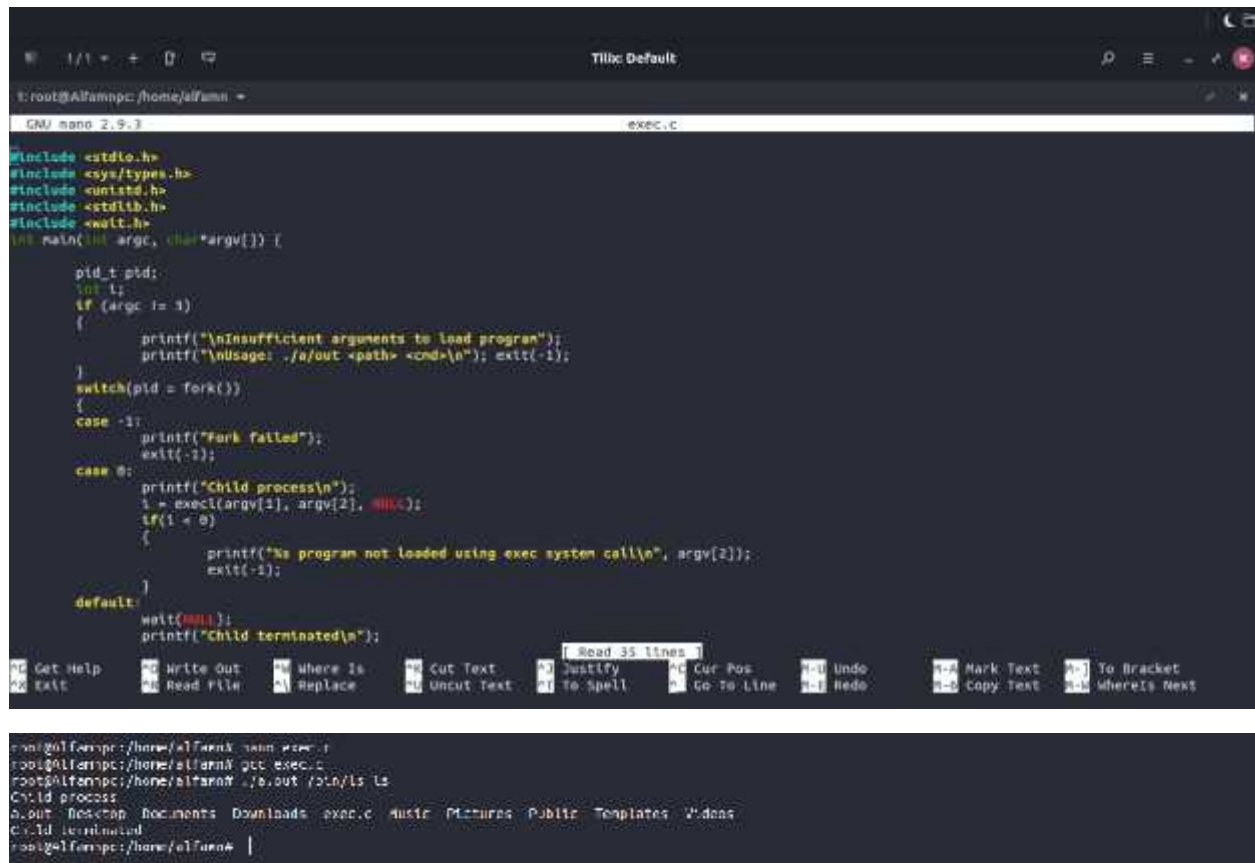


2. Wait.c

```
Terminal - kayon-pc7@kayonpc7: ~  
File Edit View Terminal Tabs Help  
wait.c  
#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("Information proses gagal!\n");  
        wait(-1);  
    }  
    else if (pid == 0) {  
        wait NULL;  
        printf("Parent starts(Nomor Ganjil):\n");  
        for (i=1; i<10; i+=2) {  
            printf("%d", i);  
            printf("\nParent ends\n");  
        }  
    }  
    else if (pid > 0) {  
        printf("Child starts(Nomor Ganjil):\n");  
        for (i=1; i<10; i+=2) {  
            printf("%d", i);  
            printf("\nChild ends\n");  
        }  
    }  
}
```

```
Terminal - kayon-pc7@kayonpc7: ~  
File Edit View Terminal Tabs Help  
}  
^  
kayon-pc7@kayonpc7:~$ nano wait.c  
kayon-pc7@kayonpc7:~$ gcc wait.c -o wait  
wait.c:28:1: error: expected identifier or '(' before '}' token  
}  
^  
kayon-pc7@kayonpc7:~$ nano wait.c  
kayon-pc7@kayonpc7:~$ gcc wait.c -o wait  
kayon-pc7@kayonpc7:~$ ./wait  
Child starts  
Nomor Ganjil: 1 3 5 7 9  
Child ends  
kayon-pc7@kayonpc7:~$ nano wait.c  
kayon-pc7@kayonpc7:~$ gcc wait.c -o wait  
kayon-pc7@kayonpc7:~$ ./wait  
Child starts  
Nomor Ganjil: 1 3 5 7 9  
Child ends  
Parent starts  
Nomor Genap: 2 4 6 8 10  
Parent ends  
kayon-pc7@kayonpc7:~$
```

3. exec.c



```
t:root@Alfarnpc:/home/alfarn#  
GNU nano 2.9.3  
exec.c  
  
#include <stdio.h>  
#include <sys/types.h>  
#include <unistd.h>  
#include <stdlib.h>  
#include <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");  
    }  
}
```

Read 35 lines

Get Help Write Out Where Is Cut Text Justify Cur Pos Undo Mark Text To Bracket
Exit Read File Replace Uncut Text To Spell Go To Line Redo Copy Text Where Is Next

```
~/Alfarnpc:/home/alfarn$ nano exec.c  
root@Alfarnpc:/home/alfarn$ gcc exec.c  
root@Alfarnpc:/home/alfarn$ ./a.out /bin/ls ls  
Child process  
./out Desktop Documents Downloads exec.c Music Pictures Public Templates Videos  
Child terminated  
root@Alfarnpc:/home/alfarn$
```