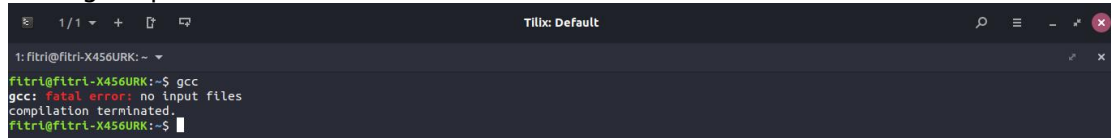


NAMA : FITRI CAHYA KUSUMAWATI
NIM : L200170110
KELAS : E

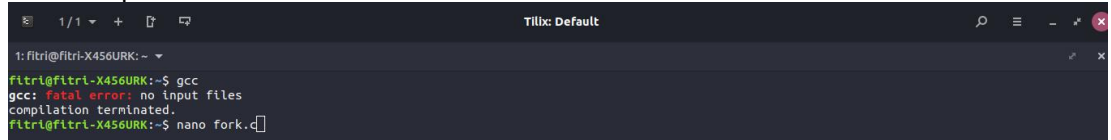
A. Ketik "gcc" pada terminal linux



```
1:fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ gcc  
gcc: fatal error: no input files  
compilation terminated.  
fitri@fitri-X456URK:~$
```

B. Fork.c

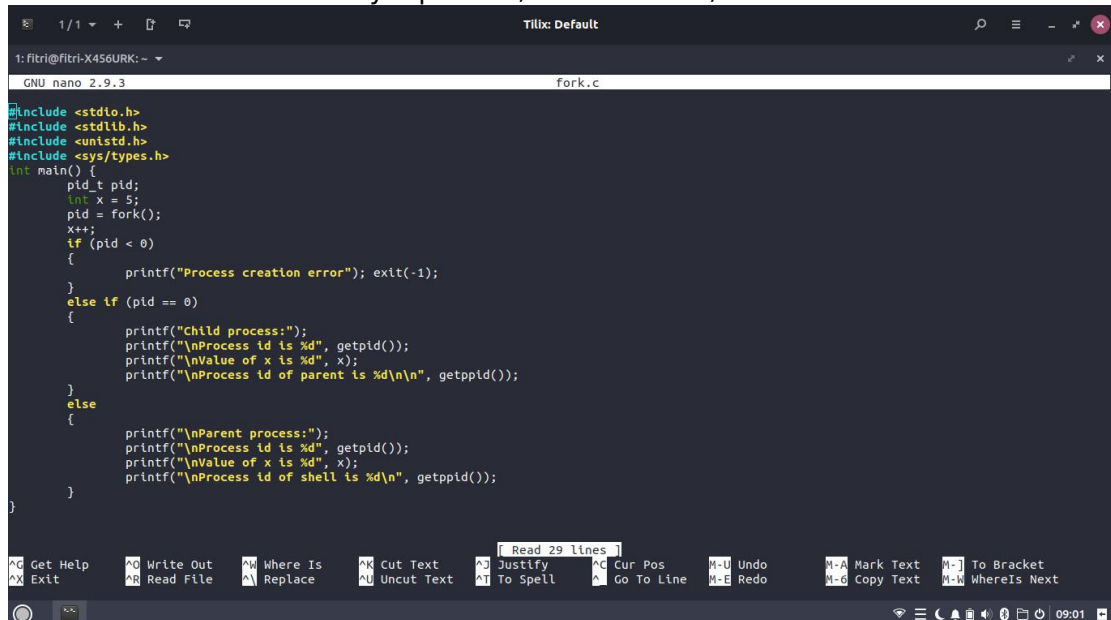
1. Ketik perintah "nano fork.c" untuk membuat file bernama fork



```
1:fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ gcc  
gcc: fatal error: no input files  
compilation terminated.  
fitri@fitri-X456URK:~$ nano fork.c
```

2. Lalu ketik perintah yang terdapat pada modul

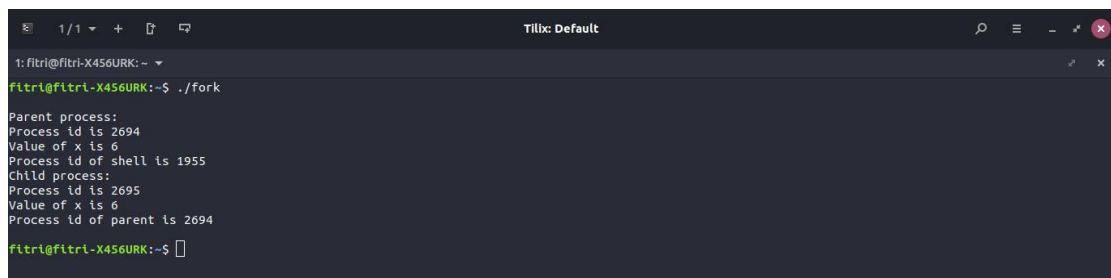
3. Tekan ctrl+x untuk menyimpan file, lalu ketik "Y", lalu enter



```
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());  
    }  
}
```

4. Ketik "gcc fork.c -o fork" untuk mengecek apakah codingan sudah benar atau belum.

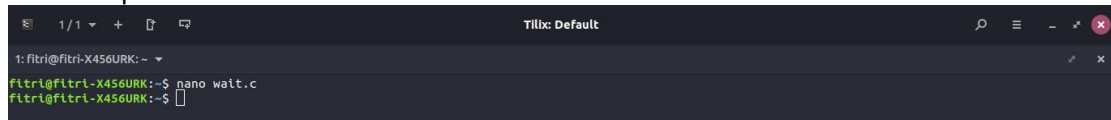
5. Ketik "./fork" untuk menampilkan outputnya



```
1:fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ ./fork  
Parent process:  
Process id is 2694  
Value of x is 6  
Process id of shell is 1955  
Child process:  
Process id is 2695  
Value of x is 6  
Process id of parent is 2694  
fitri@fitri-X456URK:~$
```

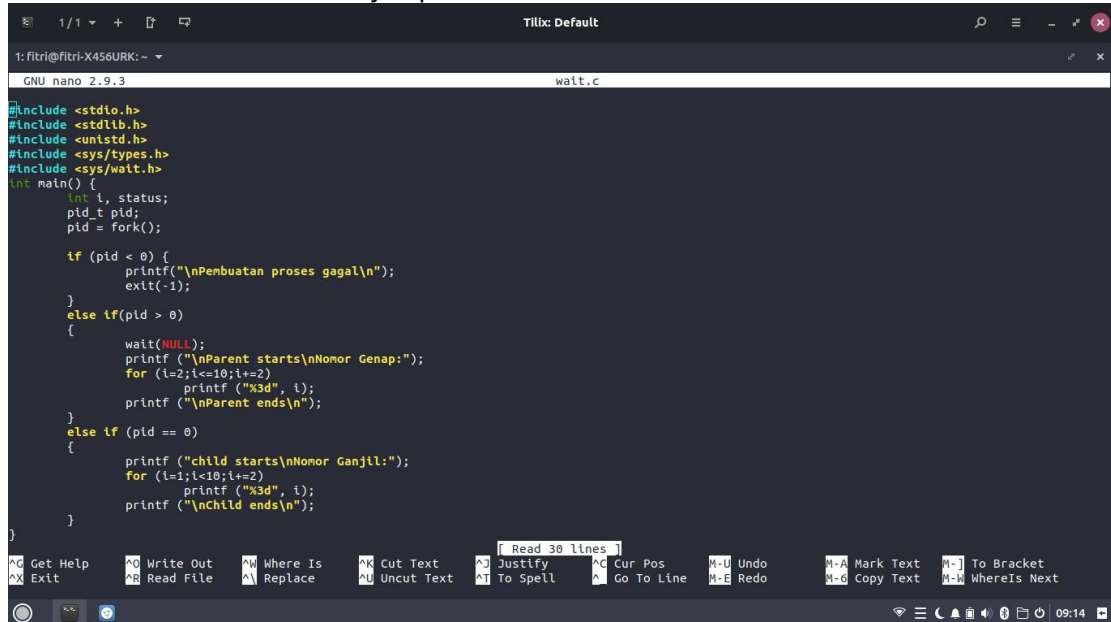
C. Wait.c

1. Ketik perintah “nano wait.c” untuk membuat file bernama wait



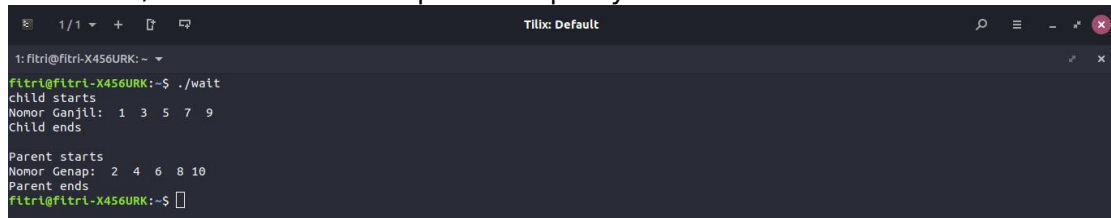
```
1: Fitri@Fitri-X456URK: ~  
Fitri@Fitri-X456URK:~$ nano wait.c  
Fitri@Fitri-X456URK:~$
```

2. Lalu ketik perintah yang terdapat pada modul
3. Tekan ctrl+x untuk menyimpan file, lalu ketik “Y”, lalu enter



```
GNU nano 2.9.3 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("\nPenbuatan 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");  
    }  
}
```

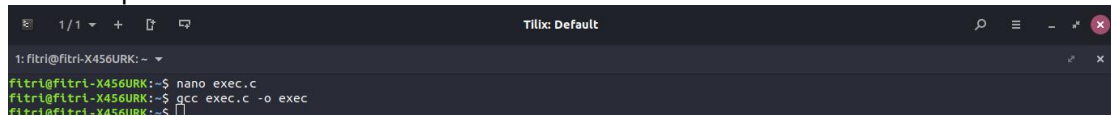
4. Ketik “gcc wait.c -o wait” untuk mengecek apakah codingan sudah benar atau belum.
5. Ketik “./wait” untuk menampilkan outputnya



```
1: Fitri@Fitri-X456URK: ~  
Fitri@Fitri-X456URK:~$ gcc wait.c -o wait  
Fitri@Fitri-X456URK:~$ ./wait  
child starts  
Nomor Ganjil: 1 3 5 7 9  
Child ends  
  
Parent starts  
Nomor Genap: 2 4 6 8 10  
Parent ends  
Fitri@Fitri-X456URK:~$
```

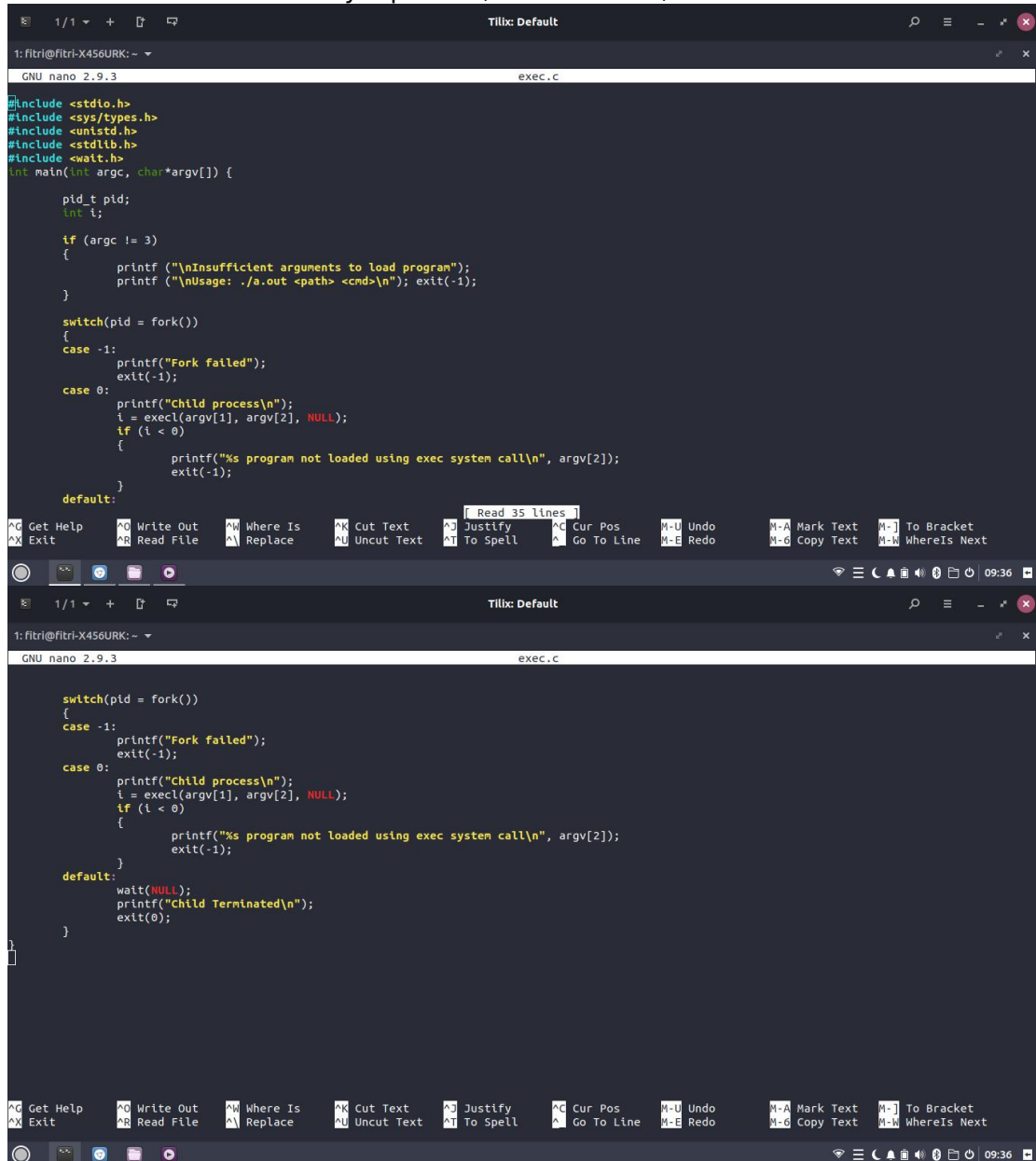
D. Exec.c

1. Ketik perintah “nano exec.c” untuk membuat file bernama exec



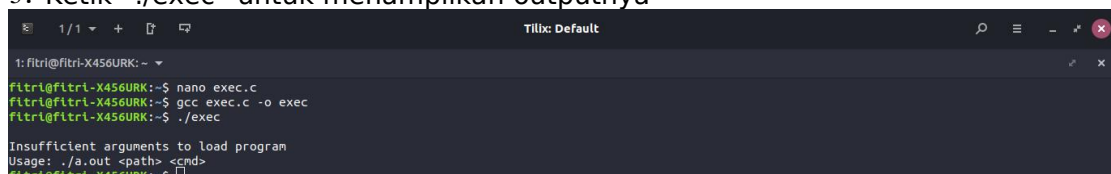
```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano exec.c  
fitri@fitri-X456URK:~$ gcc exec.c -o exec  
fitri@fitri-X456URK:~$
```

2. Lalu ketik perintah yang terdapat pada modul
3. Tekan ctrl+x untuk menyimpan file, lalu ketik “Y”, lalu enter



```
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("%s program not loaded using exec system call\n", argv[2]);  
                exit(-1);  
            }  
        default:  
            wait(NULL);  
            printf("Child Terminated\n");  
            exit(0);  
    }  
}
```

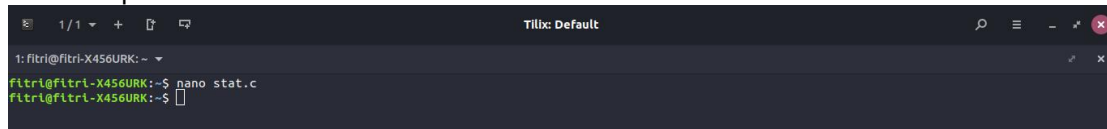
4. Ketik “gcc exec.c -o exec” untuk mengecek apakah codingan sudah benar atau belum.
5. Ketik “./exec” untuk menampilkan outputnya



```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano exec.c  
fitri@fitri-X456URK:~$ gcc exec.c -o exec  
fitri@fitri-X456URK:~$ ./exec  
Insufficient arguments to load program  
Usage: ./a.out <path> <cmd>  
fitri@fitri-X456URK:~$
```

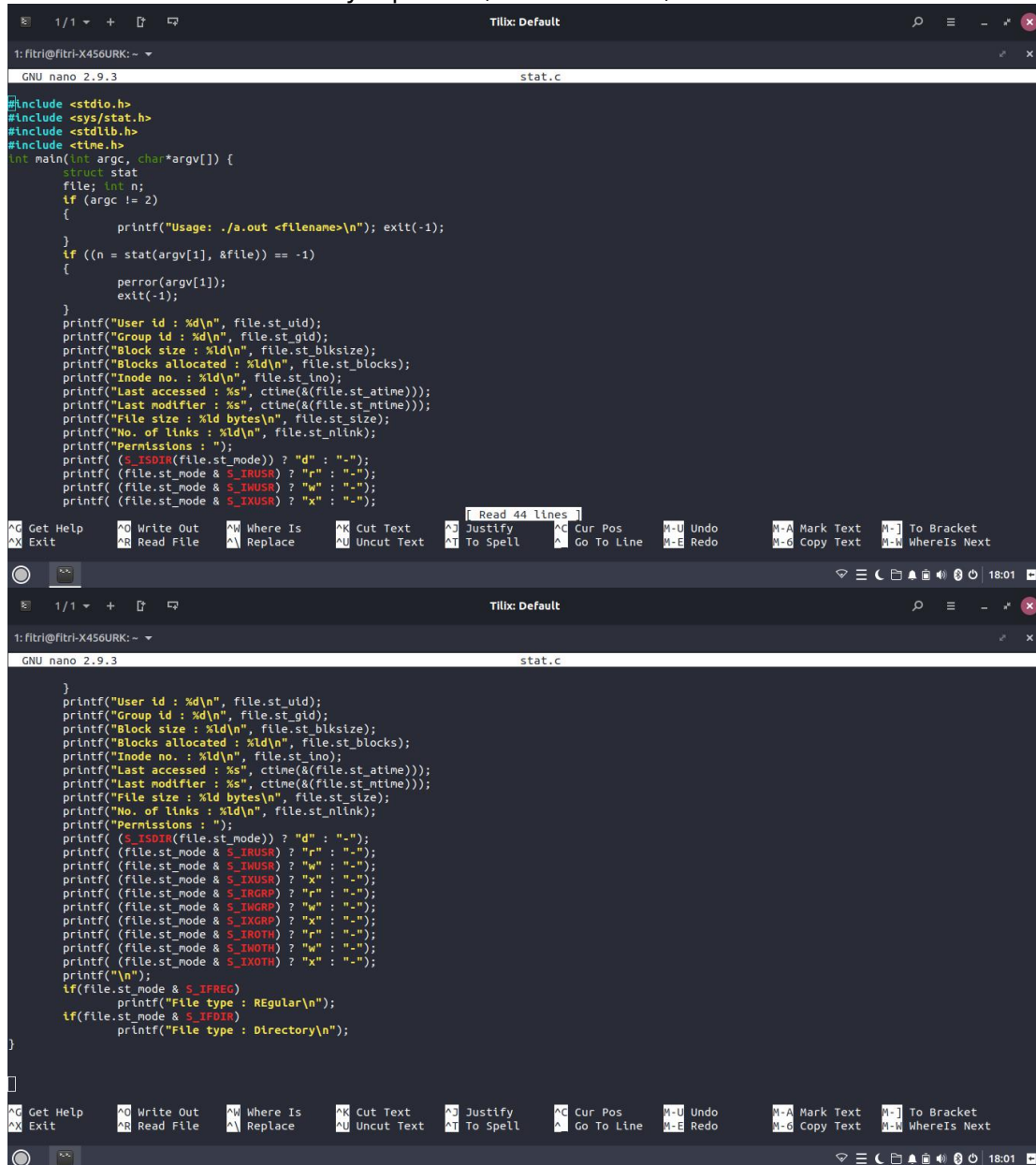
E. Stat.c

1. Ketik perintah “nano stat.c” untuk membuat file bernama stat



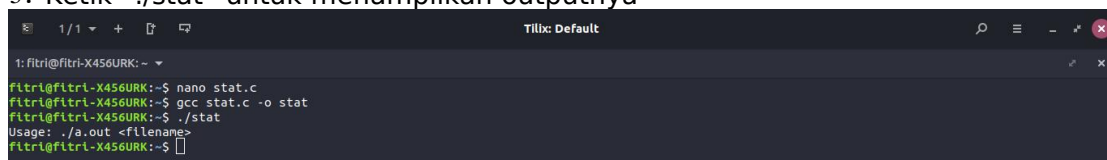
```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano stat.c  
fitri@fitri-X456URK:~$
```

2. Lalu ketik perintah yang terdapat pada modul
3. Tekan ctrl+x untuk menyimpan file, lalu ketik “Y”, lalu enter



```
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 modifier : %s", ctime(&(file.st_mtime)));  
    printf("File size : %d bytes\n", file.st_size);  
    printf("No. of links : %d\n", file.st_nlink);  
    printf("Permissions : ");  
    printf((S_ISDIR(file.st_mode)) ? "d" : "-");  
    printf((file.st_mode & S_IRUSR) ? "r" : "-");  
    printf((file.st_mode & S_IWUSR) ? "w" : "-");  
    printf((file.st_mode & S_IXUSR) ? "x" : "-");  
    printf((file.st_mode & S_IRGRP) ? "r" : "-");  
    printf((file.st_mode & S_IWGRP) ? "w" : "-");  
    printf((file.st_mode & S_IXGRP) ? "x" : "-");  
    printf((file.st_mode & S_IROTH) ? "r" : "-");  
    printf((file.st_mode & S_IWOTH) ? "w" : "-");  
    printf((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");  
}
```

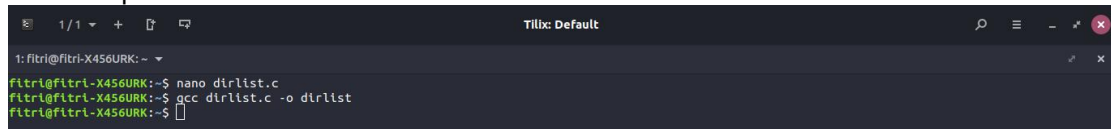
4. Ketik “gcc stat.c -o stat” untuk mengecek apakah codingan sudah benar atau belum.
5. Ketik “./stat” untuk menampilkan outputnya



```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano stat.c  
fitri@fitri-X456URK:~$ gcc stat.c -o stat  
fitri@fitri-X456URK:~$ ./stat  
Usage: ./a.out <filename>  
fitri@fitri-X456URK:~$
```

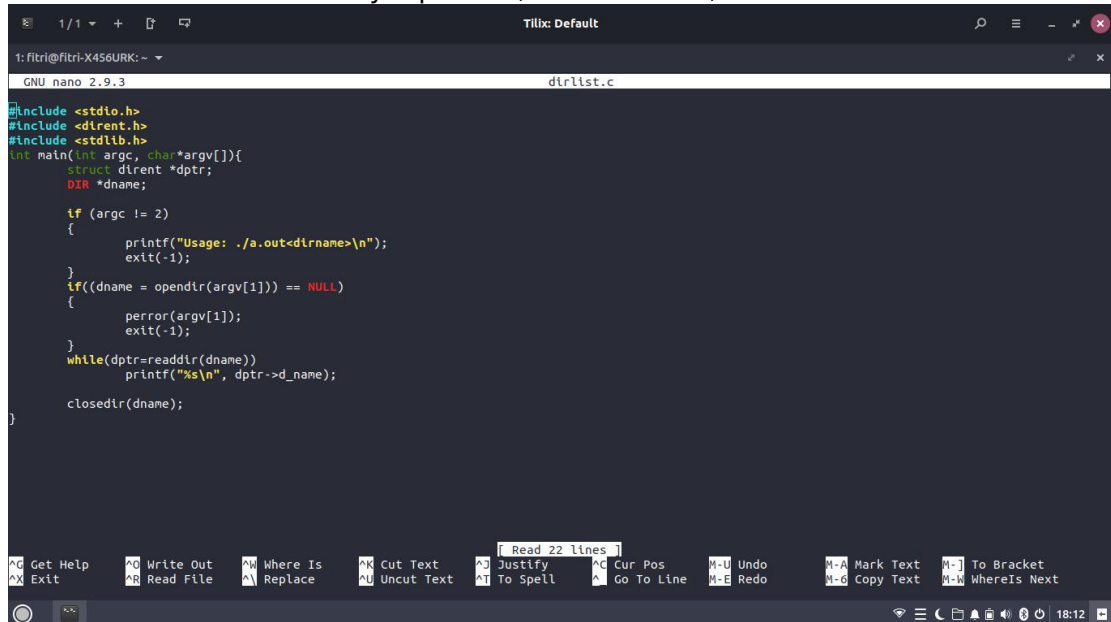
F. Dirlist.c

1. Ketik perintah “nano dirlist.c” untuk membuat file bernama dirlist



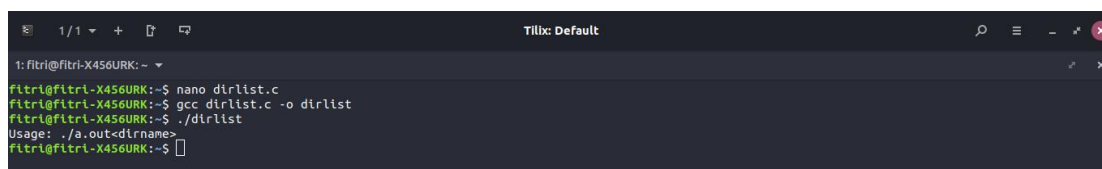
```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano dirlist.c  
fitri@fitri-X456URK:~$ gcc dirlist.c -o dirlist  
fitri@fitri-X456URK:~$
```

2. Lalu ketik perintah yang terdapat pada modul
3. Tekan ctrl+x untuk menyimpan file, lalu ketik “Y”, lalu enter



```
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);  
}
```

4. Ketik “gcc dirlist.c -o dirlist” untuk mengecek apakah codingan sudah benar atau belum.
5. Ketik “./dirlist” untuk menampilkan outputnya



```
1: fitri@fitri-X456URK: ~  
fitri@fitri-X456URK:~$ nano dirlist.c  
fitri@fitri-X456URK:~$ gcc dirlist.c -o dirlist  
fitri@fitri-X456URK:~$ ./dirlist  
Usage: ./a.out<dirname>  
fitri@fitri-X456URK:~$
```