

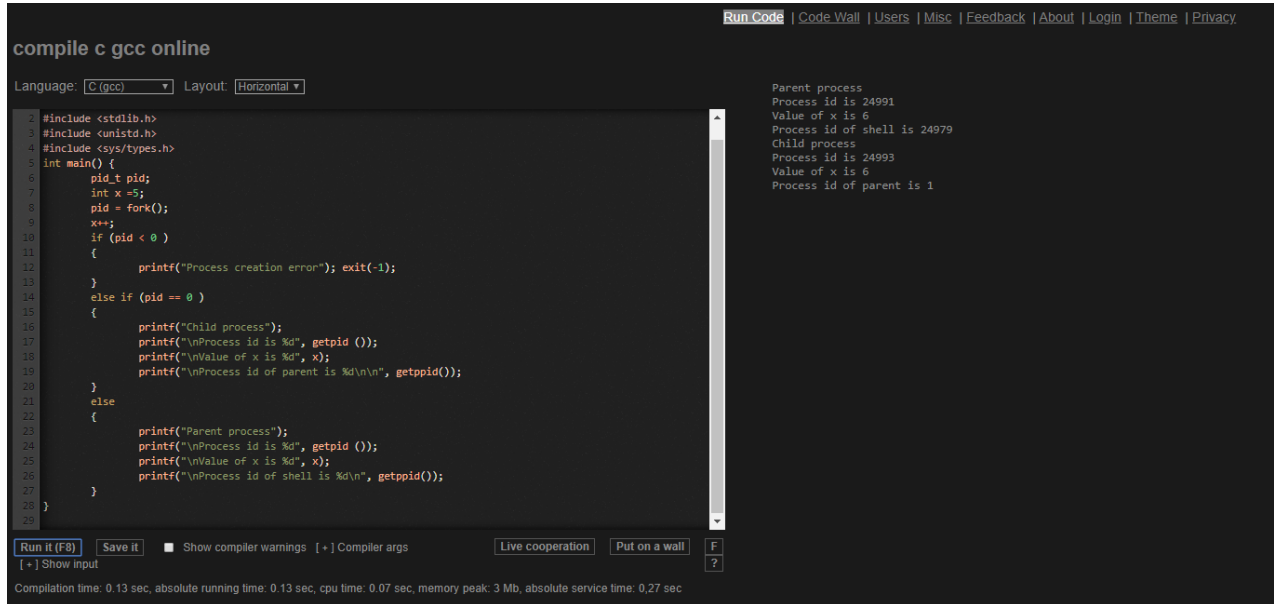
Praktikum Sistem Operasi

MODUL 8

Nama : Edi Supriyanto

NIM : L200180002

Fork.c

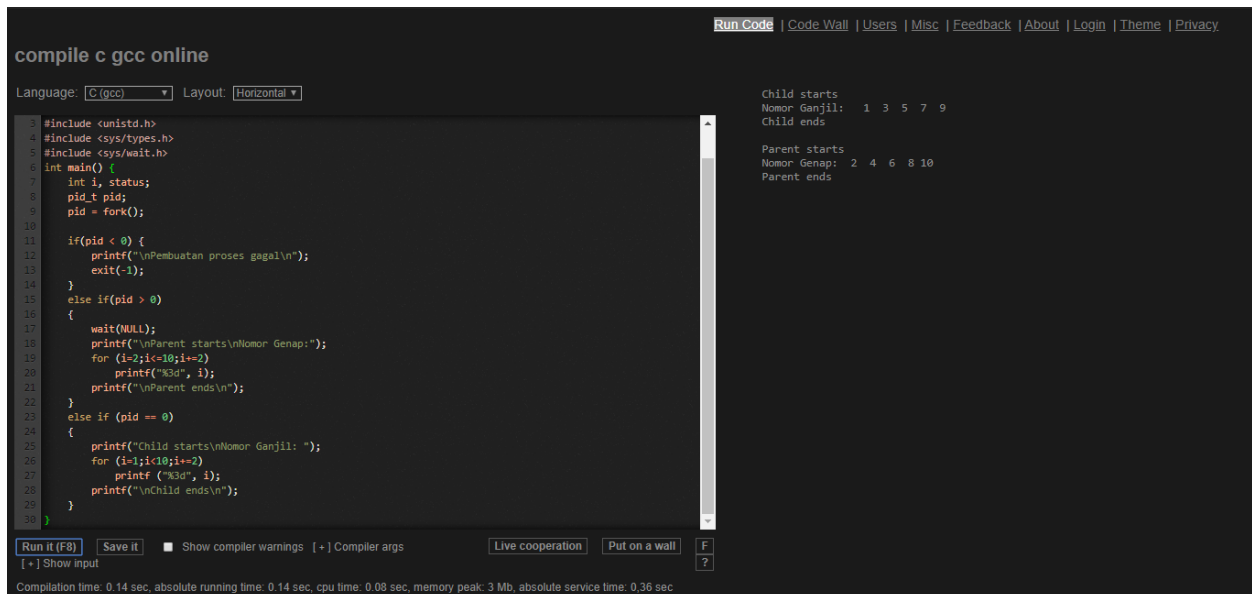


```
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 int main() {
6     pid_t pid;
7     int x = 5;
8     pid = fork();
9     x++;
10    if (pid < 0 )
11    {
12        printf("Process creation error"); exit(-1);
13    }
14    else if (pid == 0 )
15    {
16        printf("Child 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("Parent process");
24        printf("\nProcess id is %d", getpid ());
25        printf("\nValue of x is %d", x);
26        printf("\nProcess id of shell is %d\n", getppid());
27    }
28 }
```

Parent process
Process id is 24991
Value of x is 6
Process id of shell is 24979
Child process
Process id is 24993
Value of x is 6
Process id of parent is 1

Compilation time: 0.13 sec, absolute running time: 0.13 sec, cpu time: 0.07 sec, memory peak: 3 Mb, absolute service time: 0.27 sec

Wait.c



```
3 #include <unistd.h>
4 #include <sys/types.h>
5 #include <sys/wait.h>
6 int 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++)
27            printf ("%3d", i);
28        printf("\nChild ends\n");
29    }
30 }
```

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

Parent starts
Nomor Genap: 2 4 6 8 10
Parent ends

Compilation time: 0.14 sec, absolute running time: 0.14 sec, cpu time: 0.08 sec, memory peak: 3 Mb, absolute service time: 0.36 sec

Exec.c

[Run Code](#) | [Code Wall](#) | [Users](#) | [Misc](#) | [Feedback](#) | [About](#) | [Login](#) | [Theme](#) | [Privacy](#)

compile c gcc online

Language: C (gcc) Layout: Horizontal

```
1 #include <stdio.h>
2 #include <sys/types.h>
3 #include <unistd.h>
4 #include <stdlib.h>
5 int main(int argc, char*argv[]) {
6
7     pid_t pid;
8     int i;
9
10    if (argc != 3)
11    {
12        printf("\nInsufficient argument 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 = execl(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    }
```

Insufficient argument to load program
Usage: ./a.out <path> <cmd>

[Run it \(F8\)](#) [Save it](#) ☐ Show compiler warnings [+ Compiler args](#) [Live cooperation](#) [Put on a wall](#) [F](#)
[+ Show input](#) [?](#)

Compilation time: 0.13 sec, absolute running time: 0.13 sec, cpu time: 0.07 sec, memory peak: 3 Mb, absolute service time: 0.35 sec

Stat.c

[Click to go back, hold to see history](#) [Run Code](#) | [Code Wall](#) | [Users](#) | [Misc](#) | [Feedback](#) | [About](#) | [Login](#) | [Theme](#) | [Privacy](#)

compile c gcc online

Language: C (gcc) Layout: Horizontal

```
1 #include <stdio.h>
2 #include <sys/stat.h>
3 #include <stdlib.h>
4 #include <time.h>
5 int main(int argc, char*argv[]){
6     struct stat
7     file; int n;
8     if (argc != 2)
9     {
10        printf("Usage: ./a.out <filename>\n"); exit(-1);
11    }
12    if ((n = stat(argv[1], &file)) == -1)
13    {
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 accessed : %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("Permission : ");
27    printf(" (%S_ISDIR(file.st_mode)) ? \"d\" : \"-\");
28    printf(" (%S_ISUSR & S_IRUSR) ? \"r\" : \"-\"");
```

Usage: ./a.out <filename>

[Run it \(F8\)](#) [Save it](#) ☐ Show compiler warnings [+ Compiler args](#) [Live cooperation](#) [Put on a wall](#) [F](#)
[+ Show input](#) [?](#)

Compilation time: 0.13 sec, absolute running time: 0.13 sec, cpu time: 0.07 sec, memory peak: 3 Mb, absolute service time: 0.27 sec

Dirlist.c

[Run Code](#) | [Code Wall](#) | [Users](#) | [Misc](#) | [Feedback](#) | [About](#) | [Login](#) | [Theme](#) | [Privacy](#)

compile c gcc online

Language: C (gcc) Layout: Horizontal Usage: ./a.out <dirname>

```
1 #include <stdio.h>
2 #include <dirent.h>
3 #include <stdlib.h>
4 int 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 }
```

Run it (F8) Save it Show compiler warnings [+] Compiler args Live cooperation Put on a wall F ?

[+] Show input

Compilation time: 0.13 sec, absolute running time: 0.13 sec, cpu time: 0.07 sec, memory peak: 3 Mb, absolute service time: 0.34 sec