

1. Lab3 adında klasör oluşturdum. İçine dosya1 adında kaynak dosyayı oluşturdum progname.c adında kod dosyasını bu klasöre oluşturdum.
2. Programı verilen N değerleri için derleyip birleştirdim.
3. Programı N nin her değeri için birden fazla kez çalıştırdım, dosyalar hiç aynı olmadı.
4. N değerleri için sonuçlar aşağıdadır.

The screenshot shows a Kali Linux desktop environment. On the left, a text editor displays the source code for 'progname.c'. The code is a C program that creates a parent process and a child process, both of which write to files. The parent process writes to 'dosya1', 'dosya2', and 'dosya3'. The child process writes to 'dosya1', 'dosya2', and 'dosya3'. The program uses 'fork()' to create the child process and 'wait()' to wait for it to finish. The code is as follows:

```
12
13 int main (argc,argv)
14 int argc;
15 char *argv[];
16
17 {
18 int fdrd,fdr2,fdr3;
19 char chl[10];
20 char chl2[10];
21
22
23 char parent = 'P';
24 char child = 'C';
25 int pid;
26 unsigned long i;
27
28 if (argc != 1) exit (1);
29
30 if ((fdrd = open(argv[1], O_RDONLY)) == -1) exit (1);
31 if ((fdr2 = creat(argv[2], 0600)) == -1) exit (1);
32 if ((fdr3 = creat(argv[3], 0600)) == -1) exit (1);
33 printf("Parent: creating a child process\n");
34 pid = fork ();
35 if (pid == 0)
36 {
37 printf("Child process starts, id = %d\n",getpid());
38 for(;;)
39 {
40 if (read (fdrd,&chl, 1) != 1) break;
41 read (fdrd,&chl, 20);
42 for (i=0; i<10;i++) /* Uzun bir d+ng+ */
43 write (fdr2, &chl, 20);
44 write (fdr3,&chl,i);
45 }
46 exit (0);
47 }
48 else
49 {
50 printf("Parent starts, id= %d\n", getpid());
51 for(;;)
52 {
53 if (read (fdrd, &chl, 1) != 1) break;
54 read (fdrd, &chl, 10);
55 for (i=0; i<10;i++) /* Uzun bir d+ng+ */
56 write (fdr2, &chl, 10);
57 write (fdr3,&parent,i);
58 }
59 wait (0);
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
```

On the right, a file manager window shows the contents of the 'lab3' directory, which includes 'dosya1', 'dosya2', 'dosya3', and 'pro'. Below the file manager, a terminal window shows the execution of the program. The terminal output is as follows:

```
(toros@kali)~/Masaüstü/lab3
$ touch dosya1 dosya2 dosya3
(toros@kali)~/Masaüstü/lab3
$ gcc -o pro progname.c
(toros@kali)~/Masaüstü/lab3
$ ./pro progname.c
(toros@kali)~/Masaüstü/lab3
$ ./pro dosya1 dosya2 dosya3
Parent: creating a child process
Parent starts, id= 2935
PPPPPPPPPPChild process starts, id = 2936
PPPPCCPCPCCCPC
```

The screenshot shows a terminal window with the following output:

```
(toros@kali)~/Masaüstü/lab3
$ touch dosya1 dosya2 dosya3
(toros@kali)~/Masaüstü/lab3
$ gcc -o pro progname.c
(toros@kali)~/Masaüstü/lab3
$ ./pro progname.c
(toros@kali)~/Masaüstü/lab3
$ ./pro dosya1 dosya2 dosya3
Parent: creating a child process
Parent starts, id= 2935
PPPPPPPPPPChild process starts, id = 2936
PPPPCCPCPCCCPC
(toros@kali)~/Masaüstü/lab3
$
```

The screenshot shows three Notepad++ windows. The first window, titled 'dosya1', contains a list of 15 lines of text: '1 tolghah toros iAŞerik', '2 tolghah toros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 tolghah toros iAŞerik', '10 tolghah toros iAŞerik', '11 tolghah toros iAŞerik', '12 tolghah toros iAŞerik', '13 tolghah toros iAŞerik', '14 tolghah toros iAŞerik', '15 tolghah toros iAŞerik'. The second window, titled 'dosya2', contains a list of 9 lines of text: '1 tolghah toos iAŞerik', '2 tolghah oros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 to'. The third window, titled 'dosya3', contains a list of 6 lines of text: '1 iAŞerik', '2 tolghah rik', '3 tolghah toros gahan toros iAŞerik', '4 tolghah rik', '5 tolghah toros gahan toros iAŞerik', '6 gahan toros iAŞerik'.

N=5000

The screenshot shows three Notepad++ windows. The first window, titled 'dosya1', contains a list of 15 lines of text: '1 tolghah toros iAŞerik', '2 tolghah toros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 tolghah toros iAŞerik', '10 tolghah toros iAŞerik', '11 tolghah toros iAŞerik', '12 tolghah toros iAŞerik', '13 tolghah toros iAŞerik', '14 tolghah toros iAŞerik', '15 tolghah toros iAŞerik'. The second window, titled 'dosya2', contains a list of 15 lines of text: '1 tolghah toos iAŞerik', '2 tolghah oros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 tolghah toros iAŞerik', '10 tolghah toros iAŞerik', '11 tolghah toros iAŞerik', '12 tolghah toros iAŞerik', '13 tolghah toros iAŞerik', '14 tolghah toros iAŞerik', '15 i'. The third window, titled 'dosya3', contains a single line of text: '1'.

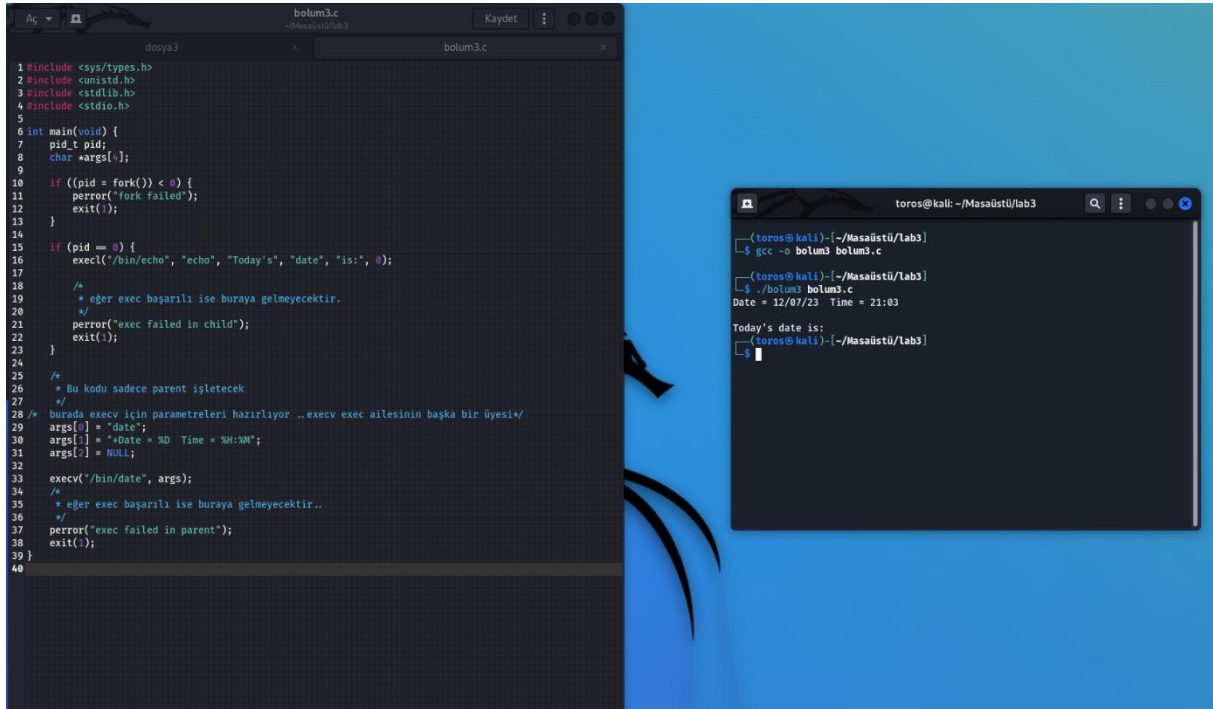
N = 10000

The screenshot shows three Notepad++ windows. The first window, titled 'dosya1', contains a list of 15 lines of text: '1 tolghah toros iAŞerik', '2 tolghah toros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 tolghah toros iAŞerik', '10 tolghah toros iAŞerik', '11 tolghah toros iAŞerik', '12 tolghah toros iAŞerik', '13 tolghah toros iAŞerik', '14 tolghah toros iAŞerik', '15 tolghah toros iAŞerik'. The second window, titled 'dosya2', contains a list of 6 lines of text: '1 tolghah toan toros ihan toros iAŞerik', '2 tolghah toros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 t', '6 tolghah n toros iAŞerik'. The third window, titled 'dosya3', contains a list of 11 lines of text: '1 os iAŞerik', '2 tolghah toros iAŞerik', '3 tolghah toros iAŞerik', '4 tolghah toros iAŞerik', '5 tolghah toros iAŞerik', '6 tolghah toros iAŞerik', '7 tolghah toros iAŞerik', '8 tolghah toros iAŞerik', '9 tolghah toros iAŞerik', '10 tolghah toros iAŞerik', '11 tolghah toros iAŞerik'.

[illegible][illegible]

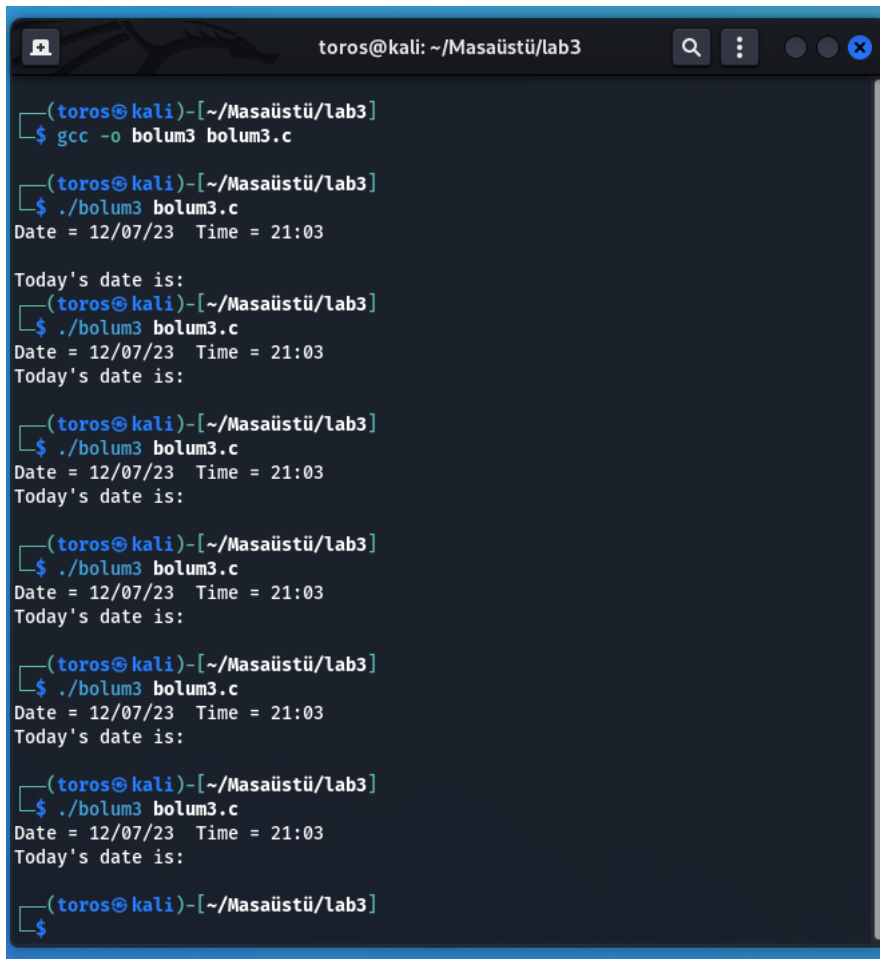
Dosya içerikleri aynı olmadı ama benzerlik oranları biraz daha arttı.

3.3) Sistem işlemi ile child prosesi mesajları karışık olarak ekrana yazdılar. Genellikle sıra doğrudur. Ama yine de bir rastlantısallıktan bahsedilebilir.



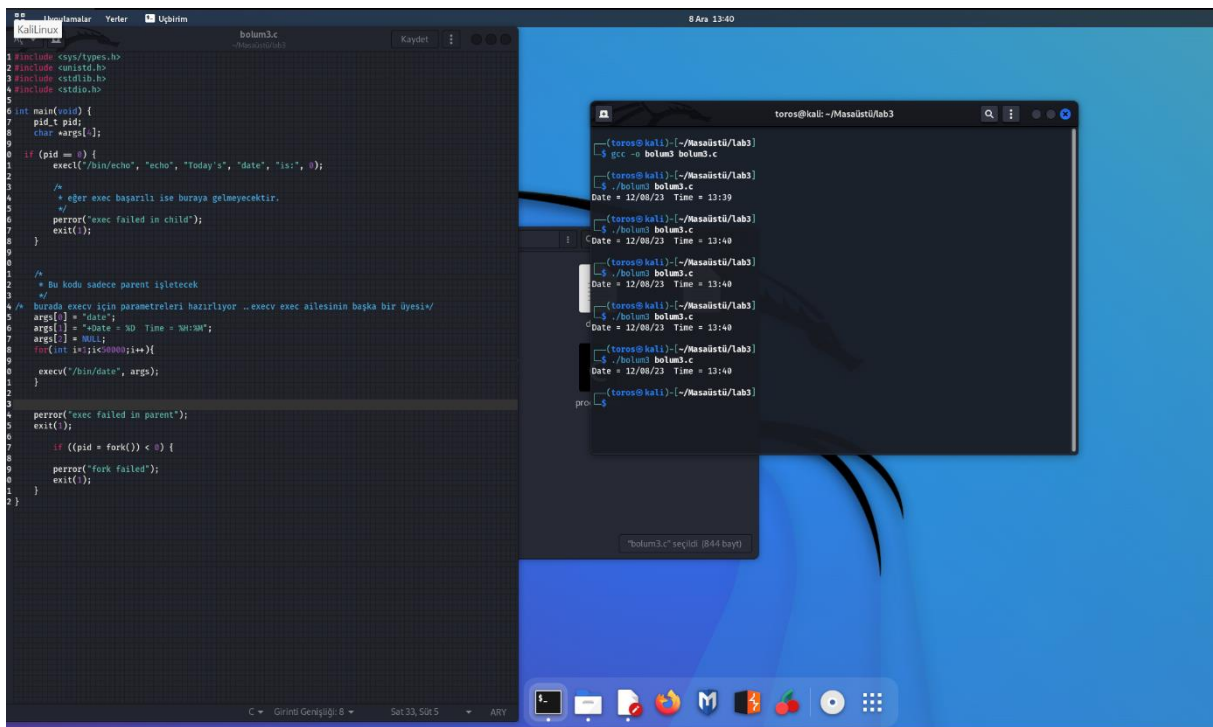
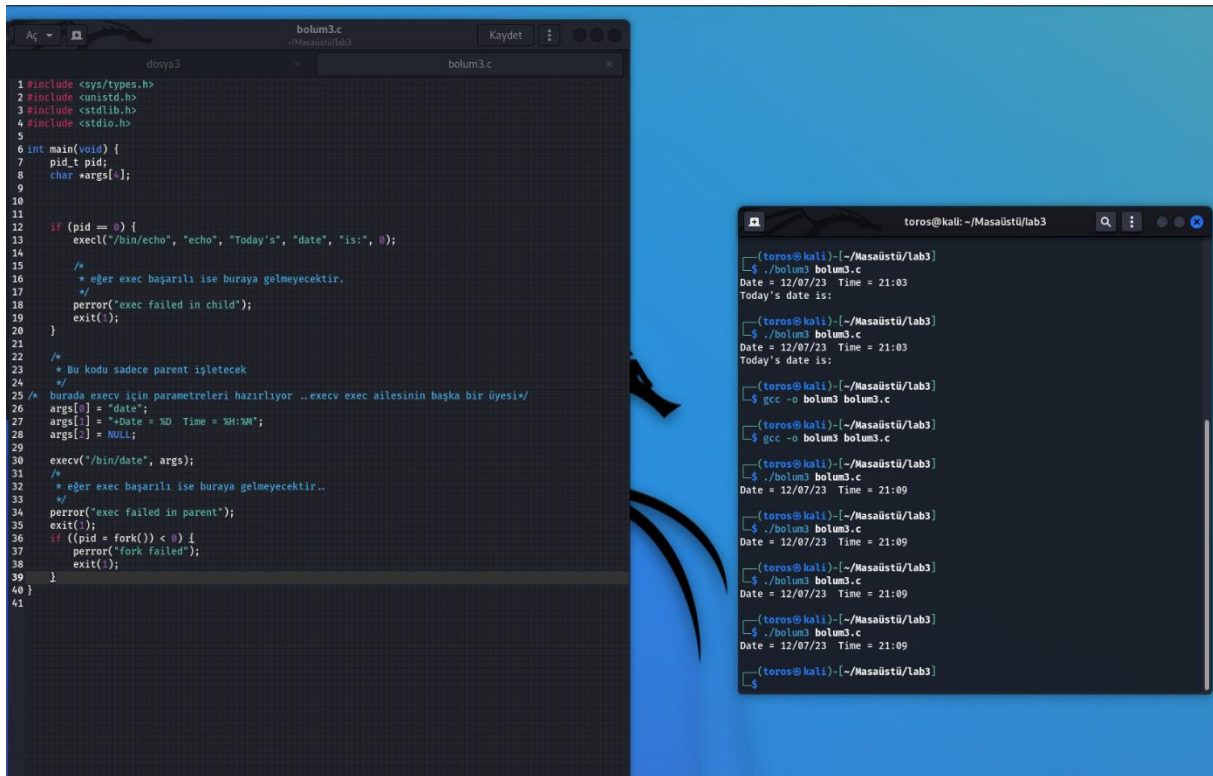
```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdlib.h>
4 #include <stdio.h>
5
6 int main(void) {
7     pid_t pid;
8     char *args[4];
9
10    if ((pid = fork()) < 0) {
11        perror("fork failed");
12        exit(1);
13    }
14
15    if (pid == 0) {
16        execl("/bin/echo", "echo", "Today's", "date", "is:", 0);
17        /*
18         * eğer exec başarılı ise buraya gelmeyecektir.
19         */
20        perror("exec failed in child");
21        exit(1);
22    }
23
24    /*
25     * Bu kodu sadece parent işletecek
26     */
27    /* burada execl için parametreleri hazırlıyor ..execl exec ailesinin başka bir üyesi*/
28    args[0] = "date";
29    args[1] = "Date = %d Time = %H:%M";
30    args[2] = NULL;
31
32    execl("/bin/date", args);
33    /*
34     * eğer exec başarılı ise buraya gelmeyecektir..
35     */
36    perror("exec failed in parent");
37    exit(1);
38 }
39
40
```

```
(toros@kali) ~/Masaüstü/lab3
$ gcc -o bolum3 bolum3.c
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$
```



```
(toros@kali) ~/Masaüstü/lab3
$ gcc -o bolum3 bolum3.c
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$ ./bolum3 bolum3.c
Date = 12/07/23 Time = 21:03
Today's date is:
(toros@kali) ~/Masaüstü/lab3
$
```


3.4) Sistem çağrısından önce for döngüsü ile bir gecikme sağlayınca mesajlar doğru sırada verildi



3.6)

execl: yeni bir işlem imajı oluşturuyor ve parametre olarak liste alıyor

execv: yeni bir işlem imajı oluşturuyor ancak parametre olarak vektör argümanlar kullanıyor