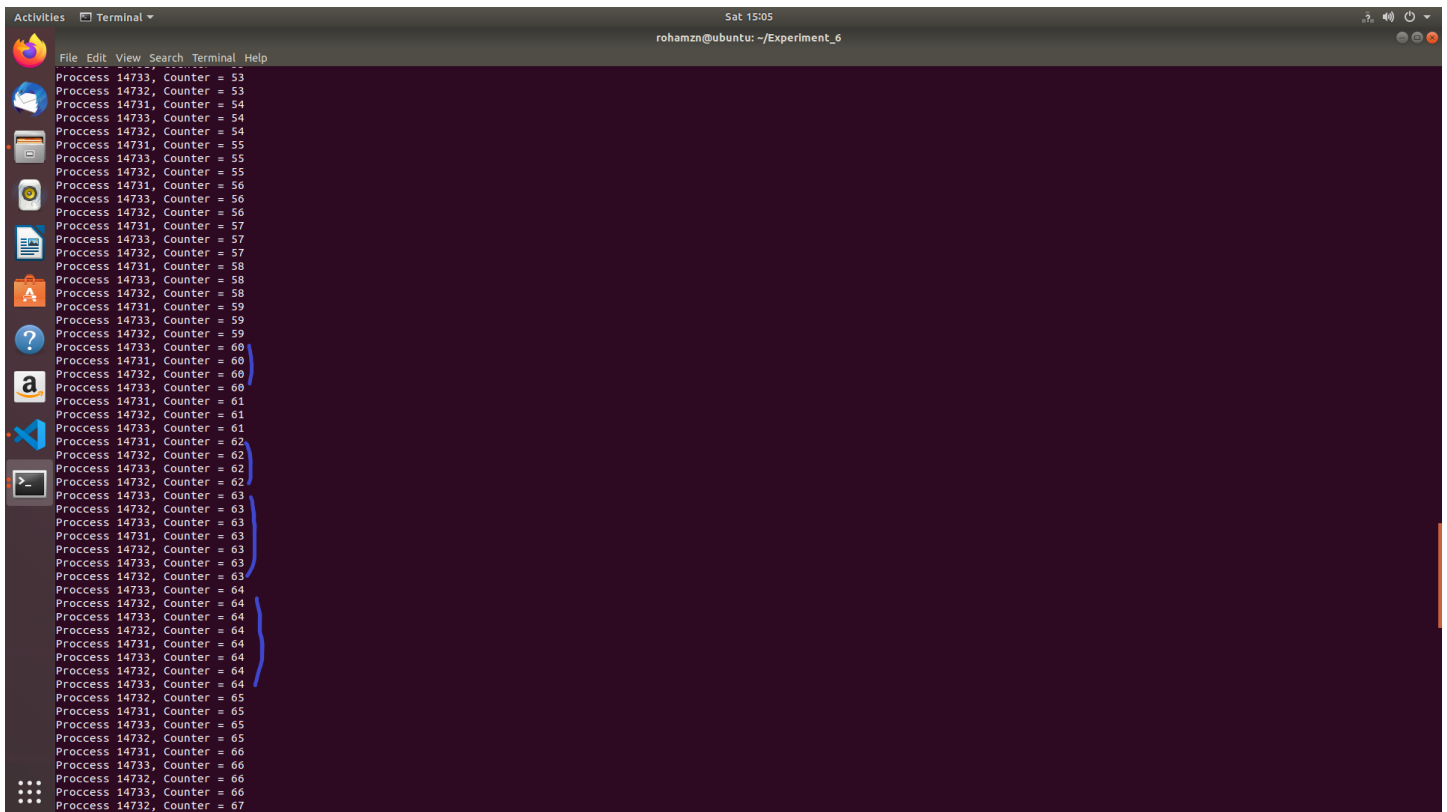


آزمایش پنجم

رهام زنده دل 9731088

بخش اول:

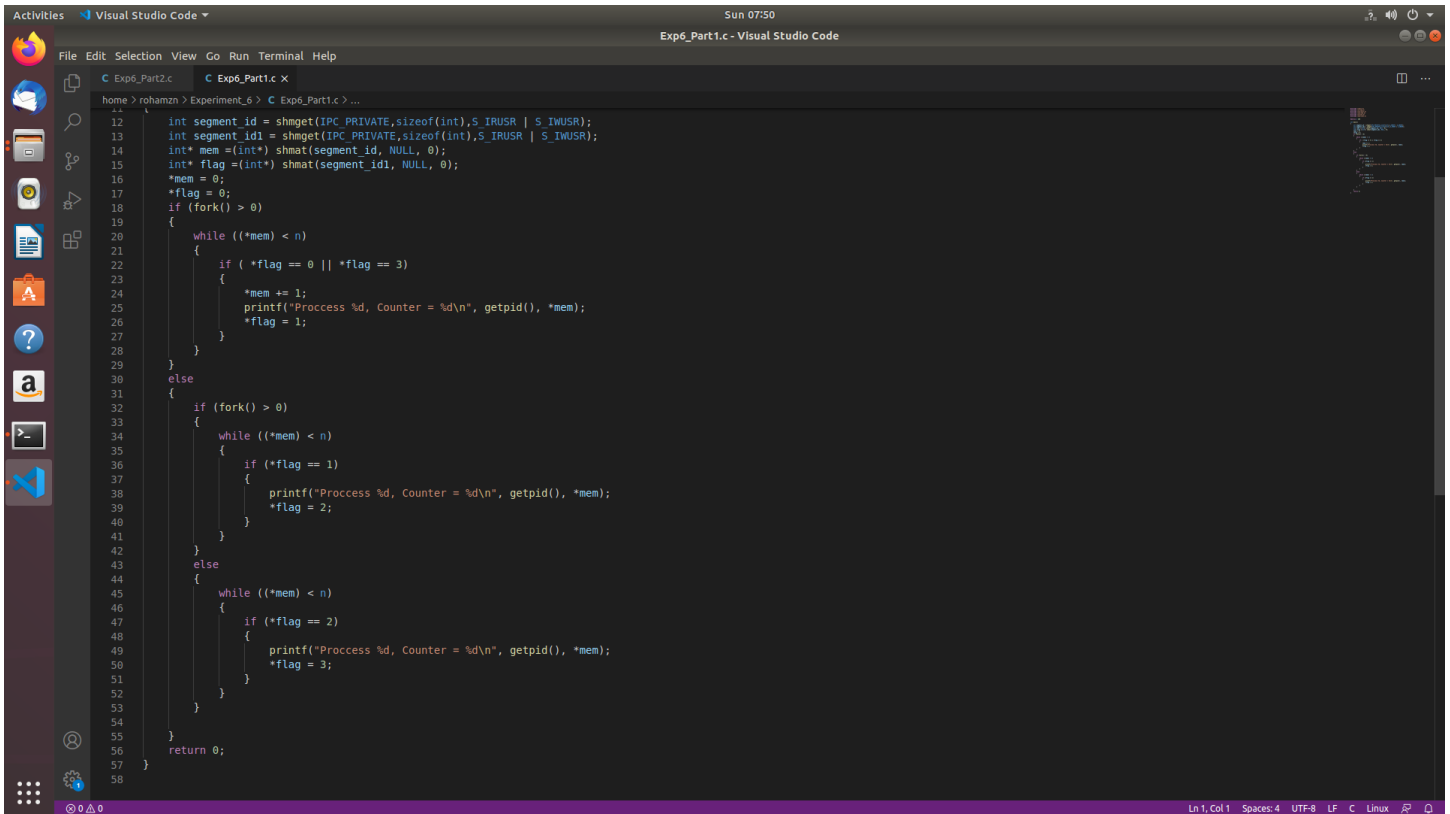


The screenshot shows a terminal window titled "Terminal" with the user "rohamzn@ubuntu: ~/Experiment_6". The terminal displays a list of processes and their counters, with a blue bracket highlighting a specific section of the output.

```
Process 14733, Counter = 53
Process 14732, Counter = 53
Process 14731, Counter = 54
Process 14733, Counter = 54
Process 14732, Counter = 54
Process 14731, Counter = 55
Process 14733, Counter = 55
Process 14732, Counter = 55
Process 14731, Counter = 56
Process 14733, Counter = 56
Process 14732, Counter = 56
Process 14731, Counter = 57
Process 14733, Counter = 57
Process 14732, Counter = 57
Process 14731, Counter = 58
Process 14733, Counter = 58
Process 14732, Counter = 58
Process 14731, Counter = 59
Process 14733, Counter = 59
Process 14732, Counter = 59
Process 14733, Counter = 60
Process 14731, Counter = 60
Process 14732, Counter = 60
Process 14733, Counter = 60
Process 14731, Counter = 61
Process 14732, Counter = 61
Process 14733, Counter = 61
Process 14731, Counter = 62
Process 14732, Counter = 62
Process 14733, Counter = 62
Process 14732, Counter = 62
Process 14733, Counter = 63
Process 14732, Counter = 63
Process 14733, Counter = 63
Process 14731, Counter = 63
Process 14732, Counter = 63
Process 14733, Counter = 63
Process 14732, Counter = 63
Process 14733, Counter = 64
Process 14732, Counter = 64
Process 14733, Counter = 64
Process 14732, Counter = 64
Process 14731, Counter = 64
Process 14733, Counter = 64
Process 14732, Counter = 64
Process 14733, Counter = 64
Process 14731, Counter = 65
Process 14732, Counter = 65
Process 14733, Counter = 65
Process 14732, Counter = 65
Process 14731, Counter = 66
Process 14733, Counter = 66
Process 14732, Counter = 66
Process 14733, Counter = 66
Process 14732, Counter = 66
Process 14733, Counter = 67
```

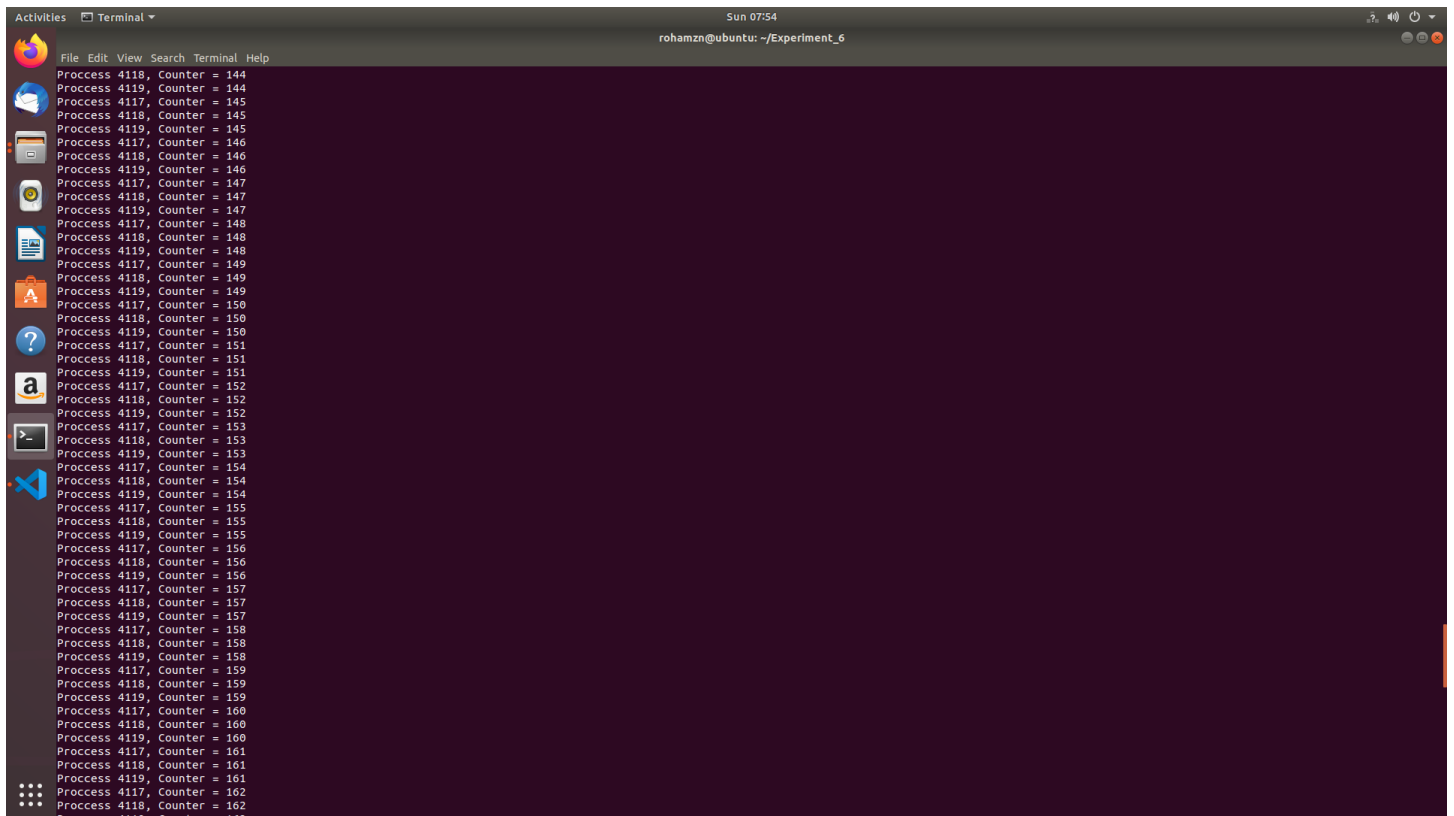
همانطور که مشاهده می شود، racing condition اتفاق می افتد.

برای رفع این مشکل از یک shared memory استفاده می کنیم که در آن وضعیت نوشته می شود.



```
12 int segment_id = shmget(IPC_PRIVATE, sizeof(int), S_IRUSR | S_IWUSR);
13 int segment_id1 = shmget(IPC_PRIVATE, sizeof(int), S_IRUSR | S_IWUSR);
14 int* mem = (int*) shmat(segment_id, NULL, 0);
15 int* flag = (int*) shmat(segment_id1, NULL, 0);
16 *mem = 0;
17 *flag = 0;
18 if (fork() > 0)
19 {
20     while ((*mem) < n)
21     {
22         if (*flag == 0 || *flag == 3)
23         {
24             *mem += 1;
25             printf("Process %d, Counter = %d\n", getpid(), *mem);
26             *flag = 1;
27         }
28     }
29 }
30 else
31 {
32     if (fork() > 0)
33     {
34         while ((*mem) < n)
35         {
36             if (*flag == 1)
37             {
38                 printf("Process %d, Counter = %d\n", getpid(), *mem);
39                 *flag = 2;
40             }
41         }
42     }
43     else
44     {
45         while ((*mem) < n)
46         {
47             if (*flag == 2)
48             {
49                 printf("Process %d, Counter = %d\n", getpid(), *mem);
50                 *flag = 3;
51             }
52         }
53     }
54 }
55 }
56 return 0;
57 }
58 }
```

Flag یک shared memory است که وضعیت آن آپدیت می شود. اگر صفر باشد یعنی در شروع برنامه هستیم. اگر 1 باشد یعنی در حال نوشتن هستیم، اگر 2 باشد یعنی اولین read و اگر 3 باشد یعنی دومین read تمام شده اند و با رسیدن به 3 دوباره شروع به write کردن می کنیم و این پروسه دوباره تکرار می شود.

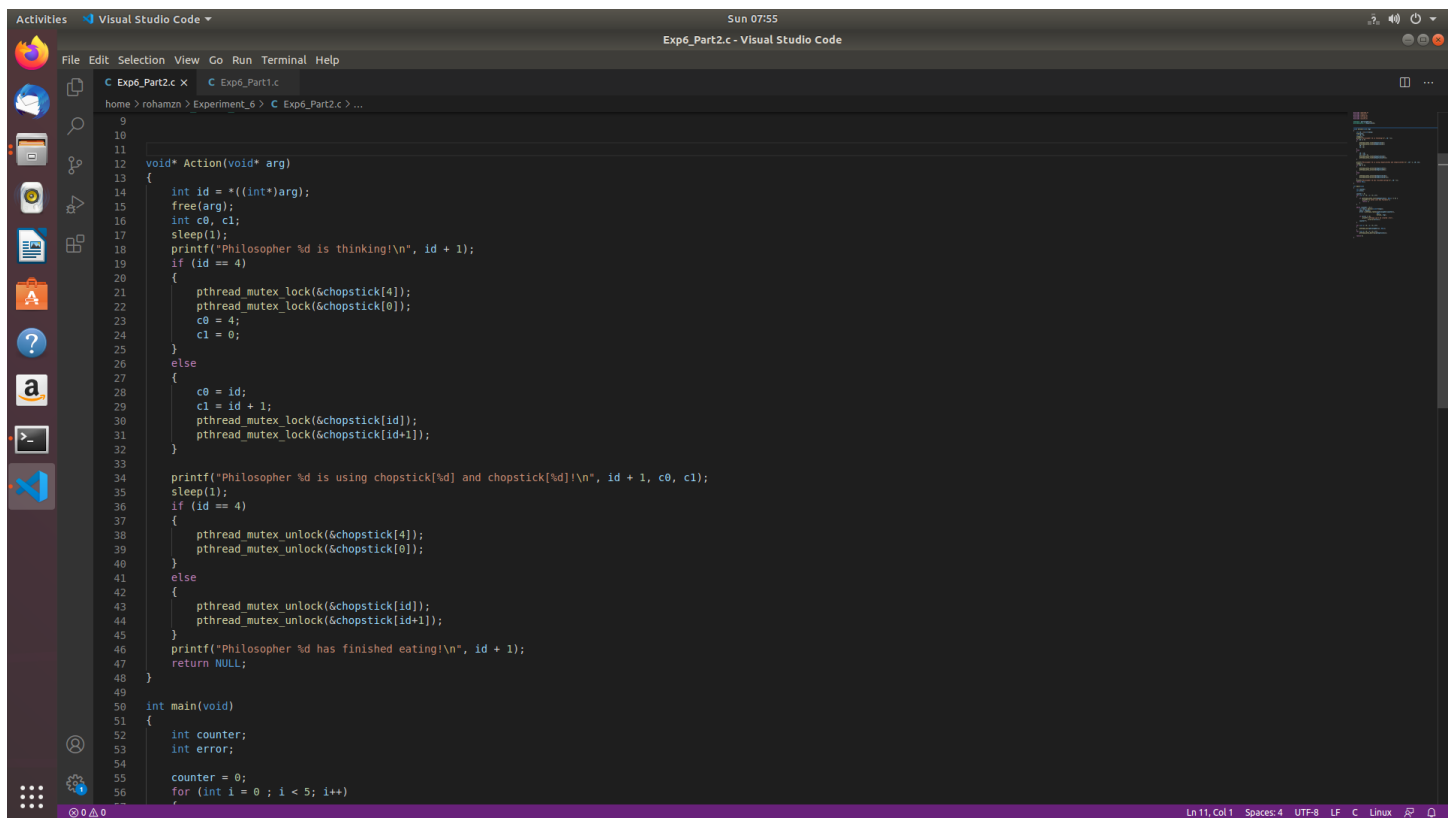


The screenshot shows a terminal window with a dark purple background. The title bar at the top indicates the user is 'rohamzn@ubuntu: ~/Experiment_6' and the time is 'Sun 07:54'. The terminal displays a list of processes and their counters, such as 'Process 4118, Counter = 144', 'Process 4119, Counter = 144', and so on, up to 'Process 4118, Counter = 162'. The list is sorted by process ID and then by counter value. The left sidebar of the terminal window shows various application icons, including a file manager, a web browser, and a terminal icon.

مشاهده می کنیم که ترتیب ها درست شده است و دیگر racing condition نداریم.

بخش 2:

هر فیلسوف برای خوردن و فکر کردن 1 ثانیه زمان مصرف می کند.



```
9
10
11
12 void* Action(void* arg)
13 {
14     int id = *((int*)arg);
15     free(arg);
16     int c0, c1;
17     sleep(1);
18     printf("Philosopher %d is thinking!\n", id + 1);
19     if (id == 4)
20     {
21         pthread_mutex_lock(&chopstick[4]);
22         pthread_mutex_lock(&chopstick[0]);
23         c0 = 4;
24         c1 = 0;
25     }
26     else
27     {
28         c0 = id;
29         c1 = id + 1;
30         pthread_mutex_lock(&chopstick[id]);
31         pthread_mutex_lock(&chopstick[id+1]);
32     }
33
34     printf("Philosopher %d is using chopstick[%d] and chopstick[%d]\n", id + 1, c0, c1);
35     sleep(1);
36     if (id == 4)
37     {
38         pthread_mutex_unlock(&chopstick[4]);
39         pthread_mutex_unlock(&chopstick[0]);
40     }
41     else
42     {
43         pthread_mutex_unlock(&chopstick[id]);
44         pthread_mutex_unlock(&chopstick[id+1]);
45     }
46     printf("Philosopher %d has finished eating!\n", id + 1);
47     return NULL;
48 }
49
50 int main(void)
51 {
52     int counter;
53     int error;
54
55     counter = 0;
56     for (int i = 0 ; i < 5; i++)
```

بعد از 1 ثانیه فکر کردن هر نخ، chopstick های سمت راست و چپ خود را lock می کند و نخ هایی که هر یک از این chopstick ها را دارند منتظر خوردن این فیلسوف می مانند. بعد از اتمام خوردن، chopstick ها unlock می شوند.

نمونه ای از اجرای بخش دوم.

```
Activities Terminal
Sun 07:39
rohamzn@ubuntu: ~/Experiment_6

rohamzn@ubuntu:~/Experiment_6$ ./Part2
Philosopher 2 is thinking!
Philosopher 3 is thinking!
Philosopher 3 is using chopstick[2] and chopstick[3]!
Philosopher 1 is thinking!
Philosopher 5 is thinking!
Philosopher 4 is thinking!
Philosopher 3 has finished eating!
Philosopher 2 is using chopstick[1] and chopstick[2]!
Philosopher 2 has finished eating!
Philosopher 1 is using chopstick[0] and chopstick[1]!
Philosopher 1 has finished eating!
Philosopher 5 is using chopstick[4] and chopstick[0]!
Philosopher 5 has finished eating!
Philosopher 4 is using chopstick[3] and chopstick[4]!
Philosopher 4 has finished eating!
rohamzn@ubuntu:~/Experiment_6$ ./Part2
Philosopher 1 is thinking!
Philosopher 1 is using chopstick[0] and chopstick[1]!
Philosopher 3 is thinking!
Philosopher 3 is using chopstick[2] and chopstick[3]!
Philosopher 2 is thinking!
Philosopher 5 is thinking!
Philosopher 4 is thinking!
Philosopher 2 is using chopstick[1] and chopstick[2]!
Philosopher 5 is using chopstick[4] and chopstick[0]!
Philosopher 1 has finished eating!
Philosopher 3 has finished eating!
Philosopher 2 has finished eating!
Philosopher 5 has finished eating!
Philosopher 4 is using chopstick[3] and chopstick[4]!
Philosopher 4 has finished eating!
rohamzn@ubuntu:~/Experiment_6$ ./Part2
Philosopher 2 is thinking!
Philosopher 5 is thinking!
Philosopher 5 is using chopstick[4] and chopstick[0]!
Philosopher 1 is thinking!
Philosopher 2 is using chopstick[1] and chopstick[2]!
Philosopher 3 is thinking!
Philosopher 4 is thinking!
Philosopher 5 has finished eating!
Philosopher 4 is using chopstick[3] and chopstick[4]!
Philosopher 2 has finished eating!
Philosopher 1 is using chopstick[0] and chopstick[1]!
Philosopher 3 is using chopstick[2] and chopstick[3]!
Philosopher 4 has finished eating!
Philosopher 1 has finished eating!
Philosopher 3 has finished eating!
rohamzn@ubuntu:~/Experiment_6$
```