

Question 2.2:

1) Show that server don't get all the signals from the client.

As we can see we sent from the client 1000 signals of SIGINT, but when we sent to the server SIGUSR1 we see that the server got only 487 SIGINT signals.

The screenshot shows a C++ IDE with a terminal window and a code editor. The terminal window is divided into two sections. The top section shows the output of the client program, and the bottom section shows the output of the server program. The code editor displays the source code for the server program, which is a simple loop that kills a process with a signal.

```
orion@Orion: ~/CLionProjects/fwork_312320062
orion@Orion:~/CLionProjects/fwork_312320062$ ./client 34985 2 1000
sig:2 , times:1000 , pid:34985 , argc:4
orion@Orion:~/CLionProjects/fwork_312320062$ ./client 34985 10 1
sig:10 , times:1 , pid:34985 , argc:4
orion@Orion:~/CLionProjects/fwork_312320062$
```

```
18 while(n<times){
19     kill(pid,sig);
20     n++;
21 }
22 return 0;
23 }
24
```

```
orion@Orion:~/CLionProjects/fwork_312320062$ ./server
My pid is: 34935
Killed
orion@Orion:~/CLionProjects/fwork_312320062$ ./server
My pid is: 34985
Server got 487 SIGINT signals
```

2) Difference between Real Time Signals and Signals.

There are few differences between RT signals and signals.
I will show some advantage and disadvantage of each method.

Standard signal disadvantage:

- signals can be ignored as we showed above that not all the signals that we sent has been received.
- We cant be sure that the signals delivered at the same order they was sent.
- Standard signals dont queue.
- The effect of signal() in a multithreaded process are unspecified.

Standard signal advantage:

- easy to use.
- Standard signals have predefined meanings.
Which means that for each signal we received we have default action to take.
- If both standard and real-time signals are pending for a process, POSIX leaves it unspecified which is delivered first. Linux, like many other implementations, gives priority to standard signals in this case.

RealTime signals advantage:

- More than one RT signal can be queued for the process if it has the signal blocked while someone send it.
- Order of delivery of RT signals is guaranteed to be the same as the sending order.
- PID and UID of sending process is written to si_pid and si_uid field of siginfo_t.
- We can pass value with the sigqueue.

RealTime signals disadvantage:

- harder to use and understand from standard signals.
- Signal queue can overflow.

- RealTime signals dont have predefined meanings. The default action for an unhandled real time singals is to terminate the receiving process.

Information taken from:

1) <https://pdfs.semanticscholar.org/69ae/2bcd1714a558465cb0264e2796937f362e98.pdf>

2) man 7 signal/man 2 signal/man signal *man sigaction* man sigqueue.

3) <https://www.linuxprogrammingblog.com/all-about-linux-signals?page=9>