Studio 7 I/O Multiplexing

1. Members

   Haiyu Wang, ID:475533

2. Select for the standard input stream

```
pi@raspberrypi-why:~/studio7 $ ./select
hello
read: hello
world
read: world
yes
read: yes
123
read: 123
```

3. Select

   Server:

```
pi@raspberrypi-why:~/studio7 $ ./server
hello
read: hello
world
read: world
adding client on fd 4
adding client on fd 5
adding client on fd 6
```

   Client:

```
pi@raspberrypi-why:~/studio7 $ ./client
Server's hostname is raspberrypi-why, current time is
19:44:29
pi@raspberrypi-why:~/studio7 $ ./client
Server's hostname is raspberrypi-why, current time is
```

```
19:44:37
pi@raspberrypi-why:~/studio7 $ ./client
Server's hostname is raspberrypi-why, current time is
19:44:38
```

4. Poll

server:

```
pi@raspberrypi-why:~/studio7 $ ./server_poll
add client fd is 4
read from client 4: hello
add client fd is 5
read from client 5: world
read from client 5: yes
read from client 4: no
hello
read from STDIN: hello
world
read from STDIN: world
quitt
read from STDIN: quitt
quit
pi@raspberrypi-why:~/studio7 $
```

client1:

```
pi@raspberrypi-why:~/studio7 $ ./client_poll
hello
no
```

client2:

```
pi@raspberrypi-why:~/studio7 $ ./client_poll
world
yes
```

5. Poll-continued

server:

```
pi@raspberrypi-why:~/studio7 $ ./server_poll
hello
read from STDIN: hello
world
read from STDIN: world
remove STDIN
hello
add client fd is 4
read from client 4: hello
read from client 4: world
```

client:

```
pi@raspberrypi-why:~/studio7 $ ./client_poll
hello
world
```

6. Epoll

a) Level-triggerd notification

```
pi@raspberrypi-why:~/studio7 $ ./epoll
hello
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
Data is available
```

```
Data is available
Data is available
Data is available
......
```

b) Edge-triggered notification

```
pi@raspberrypi-why:~/studio7 $ ./epoll
hello
Data is available
world
Data is available
yes
Data is available
oh
Data is available
```

It's quite different between these 2 modes. Since edge-triggered notification mode detect the changes in the monitored file descriptor (in this case, it's the STDIN_FILENO), it will only print "Data is available" when some changes come. Thus, there is only one message per input. However, level-triggered mode detect the exist of inputs and print messages, so there are multiple outputs.