

## Quiz 1

1. What are the differences between blocking/ non-blocking system call ? What are the difference between synchronized / asynchronized IO ? (2pts)

ans:

**Blocking vs. Synchronization**

Block/Non-block : the behavior of a function call

A blocking call returns after the requested operations complete

A non-blocking call returns ack if the system receives and starts to process the request error if the system cannot process the request

Synchronized/Asynchronized IO : the behavior of data movement

A Synchronized IO moves the data to the targeted devices and returns.

An Asynchronized IO buffers the data and moves the data to the targeted device later.

2. Assume input.txt has only one line of characters "abcdef",

```
int main() {
    int fd1, fd2;
    char buf[3];
    fd1 = open("input.txt", O_RDONLY);
    read(fd1, buf, sizeof(buf));
    write(1, buf, sizeof(buf));

    fd2 = dup(fd1);
    read(fd2, buf, sizeof(buf));
    write(1, buf, sizeof(buf));
}
```

Consider the code segment above, what will the content be in stdout after executing? (1pt)

(A) abcabc (B) abcdef (C) other

ans : B

```
int main() {
    int fd;
    char buf[3];
    pid_t pid;
    pid = fork();
    fd = open("input.txt", O_RDONLY);
    if(pid == 0) {
        /* child */
        read(fd, buf, sizeof(buf));
        write(1, buf, sizeof(buf));
    }
    else {
        /* parent will wait until child terminate*/
        int status;
        wait(&status);

        read(fd, buf, sizeof(buf));
        write(1, buf, sizeof(buf));
    }
}
```

3. Assume input.txt has only one line of characters "abcdef", Consider the code segment above, what will the content be in stdout after executing? (2pts)

(A) abcabc (B) abcdef (C) other

ans : A

Note : It forks first, then executes fd = open()