Problem 1: Read/Write

```
int main()
    int fd1, fd2;
    char c;
    fd1 = open("c.txt", O_RDONLY, 0);
    if (fork() == 0)
        read(fd1, &c, 1)
    read(fd1, &c, 1);
    printf("%c\n", c);
    exit(0);
}
```

```
Content of a.txt

12345
```

Please give all the possible output and one execution order for each. You can use line **Cx** or line **Px** to distinguish the same line of code executed by child and parent.

Problem 2: Dup

```
int main(){
    int fd1, fd2, fd3;
    char *buf1=(char*)malloc(10);
    char *buf2=(char*)malloc(10);
    memset(buf1, 0, 10);
    memset(buf2, 0, 10);
    fd1 = open("a.txt", O_RDWR, 0);
    fd2 = open("b.txt", O_RDWR|O_APPEND, 0);
    fd3 = open("a.txt", O_RDWR, 0);
```

```
Content of a.txt

abcdefg
```

```
    if(fork()==0){
        read(fd2, buf1, 2);
        dup2(fd1, fd2);
        read(fd2, buf1, 1);
        exit(0);
    }
    waitpid(-1, NULL, 0);
    read(fd2, buf1, 3);
    write(fd1, buf1, 3);
    read(fd1, buf1, 10);
    printf("%s\n", buf1);
    read(fd3, buf2, 10);
    dup2(fd2, 1);
    printf("%s\n", buf2);
    free(buf1);
    free(buf2);
    exit(0);
}
```

```
Content of b.txt:

0123456789
```

What will the contents of a.txt and b.txt be after the program completes?

What will be printed on stdout?