

## Exercise

On a UNIX system, there is a file named *data* whose stats obtained with the `ls` command are as follows:

```
i-node   blocks   permissions  owner group   bytes file
87        8        rw-r-----  peter student  7642 data
```

State the following questions:

- If `john` belongs to the same group, could `john` write the file? Can you read it? What should `peter` do to allow `john` reads the file?
- What would happen in UNIX if a user has `r--` access permission in a file and his group has `rw-?` Could the user to write the file?
- Implement a program that changes the access permissions to a UNIX file, similar to `chmod` command.

## Result

- Since the group permissions are `r--`, members of the same group can only read `peter's` files. Then `john` cannot write and read itself. So, permissions should be `r-x`, ie 101, ie 5 octal. Then you should run the command:

```
>> chmod 0650
```

- If a user has `r--` (read) access to a file and his group has `rw-` (read/write) access, the user cannot read the file, since the Access Control List (ACL) provides permissions or remove permissions to a specific user.
- We can implement a program similar to `chmod` with the code shown below, which assign all permissions to the file `"/home/hello.txt"`

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <sys/stat.h>

int main(int argc, char **argv)
{
    int mode = 0777;
    char file[100] = "/home/hello.txt";

    if (chmod (file,mode) < 0)
    {
        fprintf (stderr, "%s: error invoking chmod(%s, %o) - %d (%s)\n",
                 argv[0], file, mode, errno, strerror(errno));

        exit(1);
    }
}
```



```
return(0);  
}
```