

Rapport de Travaux Pratiques
BTS SN-IR

PSYST2 – TP1 commandes unix

Flavian LAXENAIRE

17 septembre 2021

Forme / 20	[coef. 1] →		Fond / 20	[coef. 2] →		Note / 20
Qualité du rapport			Méthodologie			
Expression écrite			Respect du Cahier des Charges			
Pertinence de la rédaction			Qualité technique			
Respect des standards de codage			État d'avancement			Malus

Sommaire

1. mycat2.....	2
1.1. 1. La commande cat et mycat.....	2
1.2. 2. Evolution de mycat.....	2
2. Mycat2.....	2
3. mycat3.....	2

1. mycat2

1.1. La commande cat et mycat

```
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$ ls
mycat  mycat2.c  mycat3.c      mycat.c  test2.c  test4.c
mycat2 mycat3      mycat3_getchar.c test1.c  test3.c
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$ ./mycat
salut
salut
comment
comment
tu vas ?
tu vas ?

snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$ ./mycat mycat.c
/* -----
| BTS SNIR 2ND ANNEE          Lycée La Fayette
+ -----
|
| Nom :          mycat.c
|
| Projet :       mycat
| Sujet :
|
| Auteur :       Flavian LAXENAIRE
| Version :
| Création :     07/09/21
| Mise à jour :
|
| Fabrication : gcc -Wall -o mycat mycat.c
+ ----- */

#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>

int main(int argc, char *argv[])
{
    char car;
    // SI ./mycat nom_fichier
    if (argc == 2)

```

```
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$ ./mycat < mycat.c
/* -----
| BTS SNIR 2ND ANNEE                Lycée La Fayette
+ -----
|
| Nom :          mycat.c
|
| Projet :       mycat
| Sujet :
|
| Auteur :       Flavien LAXENAIRE
| Version :
| Création :     07/09/21
| Mise à jour :
|
| Fabrication :  gcc -Wall -o mycat mycat.c
+ ----- */

#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>

int main(int argc, char *argv[])
{
    char car;
    // SI ./mycat nom_fichier
    if (argc == 2)
    {
        int flux;
        flux = open(*(argv + 1), O_RDWR);
        if (flux == -1)
        {
            printf("%s : ", argv[0]);
            puts("Fichier non trouvé");
        }
        else
        {
            while (read(flux, &car, 1))
            {
                write(1, &car, 1);
            }
        }
    }
}
```

```
int main(int argc, char *argv[])
{
    char car;
    // SI ./mycat nom_fichier
    if (argc == 2)
    {
        int flux;
        flux = open(*(argv + 1), O_RDWR);
        if (flux == -1)
        {
            printf("%s : ", argv[0]);
            puts("Fichier non trouvé");
        }
        else
        {
            while (read(flux, &car, 1))
            {
                write(1, &car, 1);
            }
            close(flux);
            putchar('\n');
            return 0;
        }
    }

    // SI ./mycat
    else if (argc == 1)
    {
        while (read(0, &car, 1))
        {
            write(1, &car, 1);
        }
        putchar('\n');
        return 0;
    }
}
```

2. Mycat2

```
int lire_fichier(char *nom, char *buffer)
{
    char car;
    // SI ./mycat nom_fichier
    int octet = 0;
    int flux;
    flux = open(nom, O_RDONLY);
    if (flux == -1)
    {
        puts("Fichier non trouvé");
        return -1;
    }
    else
    {
        while (read(flux, &car, 1))
        {
            //write(1, &car, 1);
            buffer[0]=car;
            octet++;
            printf("%c", buffer[0]);
        }
        close(flux);
        printf("%s", buffer);
        putchar('\n');
        printf("%d", octet);
        return octet;
    }
}
```

3. mycat3

```
int main(int argc, char *argv[])
{
    char car;
    char car2;
    int octet = 0;
    //printf("%d\n", argc);
    int n = 1;
    int arguments = argc;
    char buffer[10];
    if (argc > 1)
    {
        while (arguments > 1)
        {
            if (strcmp(argv[n], "-") == 0)
            {
                while (read(0, &car2, 1) != 0)
                    lire_fichier(*(argv + n + 1), buffer);
            }
            else
            {
                lire_fichier(*(argvn", arguments);
                n++;
                arguments--;
            }
        }
        } + n), buffer);
        // printf("argc%d_ \n", arguments);
        n++;
        arguments--;
    }
}
```

```
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$ ./mycat3 test1.c - test2.c - test3.c - test4.c
salut
comment
tu vas
?
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myCat$
```

4. La commande ls et myls

Ouvrir le répertoire courant ou un dossier spécifique :

```
char nom[25];
strcpy(nom, ".");

if (argc == 1)
    strcpy(nom, ".");
else if (argc == 2 && (strcmp(argv[1], "-l") != 0))
    strcpy(nom, argv[1]);
else if (argc == 3)//option -l à priori ...
{
    strcpy(nom, argv[2]);
}
```

tester si le nom passé en argument est un répertoire :

```
if ((bufstat.st_mode & S_IFMT) != S_IFDIR)
    printf("%s: %s is not a directory \n", argv[0], nom);
```

Si il est lisible :

```
if ((fd = open(nom, O_RDONLY)) == -1)
    printf("%s: cannot read \n", argv[0]);
```

La fonction mode() complétée :

```
if ((bufstat2.st_mode & S_IRWXU) == S_IRWXU)
{ //OWNER
    //printf("acces total user");
    p += 1;
    *p++ = 'r';
    *p++ = 'w';
    *p = 'x';
    p -= 3;
}
if ((bufstat2.st_mode & S_IRUSR) == S_IRUSR)
{
    p += 1;
    *p = 'r';
    p--;
}
if ((bufstat2.st_mode & S_IWUSR) == S_IWUSR)
{
    p += 2;
    *p = 'w';
    p -= 2;
}
if ((bufstat2.st_mode & S_IXUSR) == S_IXUSR)
{
    p += 3;
    *p = 'x';
    p -= 3;
}
```

```

if ((bufstat2.st_mode & S_IRWXG) == S_IRWXG)
{ //GROUPE
    p += 4;
    *p++ = 'r';
    *p++ = 'w';
    *p = 'x';
    p -= 6;
}
if ((bufstat2.st_mode & S_IRGRP) == S_IRGRP)
{
    p += 4;
    *p = 'r';
    p -= 4;
}
if ((bufstat2.st_mode & S_IWGRP) == S_IWGRP)
{
    p += 5;
    *p = 'w';
    p -= 5;
}
if ((bufstat2.st_mode & S_IXGRP) == S_IXGRP)
{
    p += 6;
    *p = 'x';
    p -= 6;
}
//OTHER
if ((bufstat2.st_mode & S_IRWXO) == S_IRWXO)
{ //AUTRE
    p += 7;
    *p++ = 'r';
    *p++ = 'w';
    *p = 'x';
    p -= 9;
}
if ((bufstat2.st_mode & S_IROTH) == S_IROTH)
{
    p += 7;
    *p = 'r';
    p -= 7;
}
if ((bufstat2.st_mode & S_IWOTH) == S_IWOTH)
{
    p += 8;
    *p = 'w';
    p -= 8;
}
if ((bufstat2.st_mode & S_IXOTH) == S_IXOTH)
{
    p += 9;
    *p = 'x';
    p -= 9;
}
}

```

le retour de la fonction mode lorsque l'argument -l est passé :

```

char * time = ctime(&bufstat2.st_atim.tv_sec);
if(time[strlen(time)-1] == '\n') time[strlen(time)-1] = '\0';
printf(" %s %d %ld %s ", mod, bufstat2.st_uid, bufstat2.st_size, time);
return 1;

```

```
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myLs$ ls
myls myls.c test
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myLs$ ./myls
myls.c
test
..
myls
.
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myLs$ ./myls -l
-rw-rw-r-- 1000 5673 Fri Sep 17 08:37:02 2021 myls.c
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:50 2021 test
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:35 2021 ..
-rwxrwxr-x 1000 13064 Fri Sep 17 08:37:37 2021 myls
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:50 2021 .
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myLs$ ./myls -l test/
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:50 2021 test
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:35 2021 ..
drwxrwxr-x 1000 4096 Fri Sep 17 08:34:50 2021 .
snir@snir-21-22:~/Documents/bts snir 2nd annee/dev/psyst2/tp1/myLs$
```