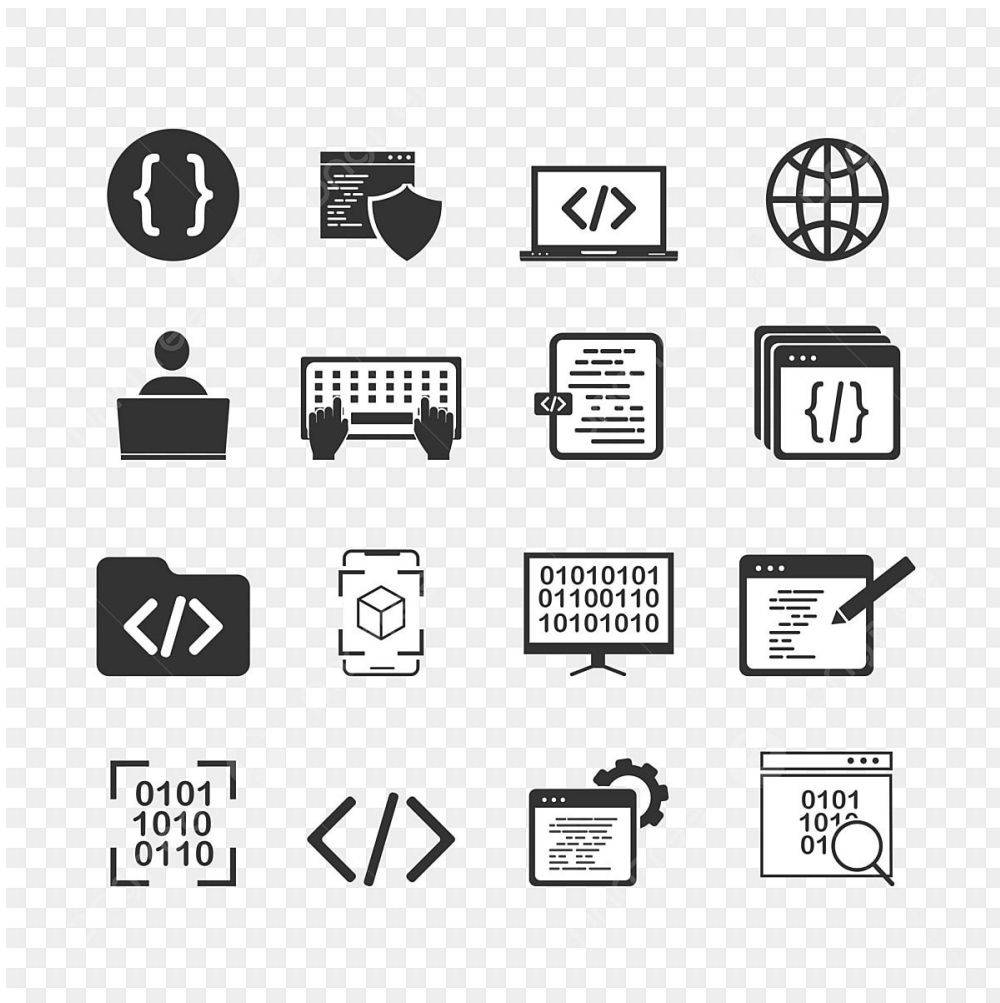


Guide de l'utilisateur

pour la bibliothèque de fonctions de gestion de fichiers du
projet d'O.S.

par M'GHAIEETH Fayçal, KOTOKPO Josué, NASRO Rona.

Mars 2024.



1. Paramètres

Pour utiliser la bibliothèque, il faut lancer le terminal UNIX, se placer dans le fichier projet et taper **./function**, comme ci-dessous :

```
jojo@LAPTOP-DLN:~/project$ ./function

Manuel de test:
1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage mySeek(f, 2, SEEK_CUR);
6. Lire en faisant un décallage mySeek(f, 2, SEEK_SET);
7. Lire en faisant un décallage mySeek(f, 2, SEEK_END);
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter


Votre choix: █
```

Plusieurs choix sont alors disponibles selon les différentes fonctionnalités proposées.

Cas 1 : En tapant 1, l'utilisateur peut créer et formater la partition :

```
Manuel de test:
1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage mySeek(f, 2, SEEK_CUR);
6. Lire en faisant un décallage mySeek(f, 2, SEEK_SET);
7. Lire en faisant un décallage mySeek(f, 2, SEEK_END);
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Votre choix: 1
Partition créée ou formatée avec succès.
```



Cas 2 : En tapant 2, l'utilisateur peut ouvrir un fichier :

```
Votre choix: 2
Fichier 'fichierDeTest.txt' ouvert avec succès.
```

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 2, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Cas 3 : En tapant 3, l'utilisateur peut écrire dans un fichier :

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 2, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Votre choix: 3

Entrez les données à écrire dans le fichier : Bonjour CY Universite
Écriture réussie : 21 octets écrits.

Cas 4 : En tapant 4, l'utilisateur peut lire un fichier :

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 2, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Votre choix: 4

Contenu du fichier : Bonjour CY Universite

Cas 5 : En tapant 5, l'utilisateur peut lire en faisant un décallage `mySeek(f, 2, SEEK_CUR)` :

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 2, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Votre choix: 5

Contenu du fichier : njour CY Universite

Cas 6 : En tapant 6, l'utilisateur peut lire en faisant un décallage `mySeek(f, 2, SEEK_SET)`.

Cas 7 : En tapant 7, l'utilisateur peut lire en faisant un décallage `mySeek(f, 2, SEEK_END)`.

Cas 8 : En tapant 8, l'utilisateur peut obtenir la taille d'un fichier :

```
Votre choix: 8
Taille du fichier 'fichierDeTest.txt' : 21 octets
```

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 0, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Cas 9 : En tapant 9, l'utilisateur peut visualiser l'espace occupé dans la partition :

```
Votre choix: 9
Espace utilisé : 2048 o / 32 Mo
```

Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décallage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décallage `mySeek(f, 0, SEEK_SET);`
7. Lire en faisant un décallage `mySeek(f, 2, SEEK_END);`
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
0. Quitter

Cas 0 : En tapant 0, l'utilisateur peut quitter l'application.


2. Fichiers de configuration

Ci-dessous les fichiers de configuration nécessaires au bon fonctionnement du projet :

- **function.h** : Dans ce fichier se trouvent la déclaration de la structure représentant les métadonnées d'un fichier, les variables globales utilisées, ainsi que la déclaration des différentes fonctions :

- **myFormat** : formatage de la partition
 - **myOpen** : ouverture d'un fichier
 - **visualisation** : visualiser la partition avec les parties réservées et libres
 - **mySeek** : déplacement du curseur
 - **myRead** : lecture des données d'un fichier
 - **myWrite** : écriture des données dans un fichier
 - **size** : taille d'un fichier
- **makefile** : Pour lancer le makefile nécessaire à la bonne compilation du programme, l'utilisateur doit se placer dans le dossier projet et taper la commande **make** puis **./projet** pour lancer :

```
jojo@LAPTOP-DLN:~/project$ make
make: 'projet' is up to date.
jojo@LAPTOP-DLN:~/project$ ./projet
```



Manuel de test:

1. Créer ou formater la partition
2. Ouvrir un fichier
3. Écrire dans un fichier
4. Lire un fichier
5. Lire en faisant un décalage `mySeek(f, 2, SEEK_CUR);`
6. Lire en faisant un décalage `mySeek(f, 0, SEEK_SET);`
7. Écrire/Créer un nouveau fichier autre que notre fichier de test;
8. Taille d'un fichier
9. Visualiser l'espace occupé dans la partition
10. Écrire en faisant un décalage `mySeek(f, 2, SEEK_CUR);`
11. Écrire en faisant un décalage `mySeek(f, 0, SEEK_SET);`
0. Quitter