[C] Opérations systèmes

Introduction

Le langage C permet d'effectuer des opérations systèmes grâce à certaines fonctions de bibliothèques.

Création d'un répertoire

La fonction **mkdir** permet de créer un répertoire :

```
#include <stdio.h>
#include <sys/stat.h>

int main() {
    const char* nom_du_repertoire = "nouveau_repertoire";

    if (mkdir(nom_du_repertoire, 0777) == 0) {
        printf("Répertoire créé avec succès.\n");
    } else {
        printf("Erreur lors de la création du répertoire.\n");
    }

    return 0;
}
```

Affichage du contenu d'un répertoire

La fonction **readdir** prend le chemin d'un répertoire en entrée et retourne un tableau avec les chemins de chaque sous-dossiers :

```
#include <stdio.h>
#include <dirent.h>

int main() {
    const char* chemin_du_dossier = "."; // Chemin du dossier à lister, ici le dossier courant

    DIR* directory = opendir(chemin_du_dossier);
    struct dirent* file;

    if (directory) {
        while ((file = readdir(directory)) != NULL) {
            printf("%s\n", file->d_name);
        }
        closedir(directory);
    } else {
        printf("Erreur lors de l'ouverture du répertoire.\n");
    }

    return 0;
}
```

Affichage du chemin du répertoire courant

Vous pouvez récupérer le CWD (Current Working Directory) grâce à la fonction **getcwd** :

```
#include <stdio.h>
#include <unistd.h>

int main() {
    char buffer[1024]; // Buffer pour stocker le chemin

    if (getcwd(buffer, sizeof(buffer)) != NULL) {
```

```
        printf("Le répertoire de travail actuel est : %s\n", buffer);
    } else {
        perror("Erreur lors de l'obtention du répertoire de travail");
        return 1;
    }

    return 0;
}
```

Affichage d'information d'un fichier

Grâce à la commande **stat** vous allez pouvoir afficher les caractéristiques d'un fichier tel que sa taille, son type, ses permissions, etc.

```
#include <stdio.h>
#include <sys/stat.h>
#include <unistd.h>

int main() {
    const char *nom_fichier = "mon_fichier.txt"; // Le nom du fichier que vous voulez analyser

    struct stat info;
    if (stat(nom_fichier, &info) == 0) {
        printf("Nom du fichier : %s\n", nom_fichier);
        printf("Taille du fichier : %ld octets\n", info.st_size);
        printf("Permissions : %o\n", info.st_mode & 0777); // Affichage des permissions en
octal
        // Vous pouvez accéder à plus d'informations dans la structure 'info'
    } else {
        perror("Erreur lors de la récupération des informations du fichier");
        return 1;
    }

    return 0;
}
```

Renommer un fichier

Vous allez pouvoir renommer vos fichier grâce à la fonction **rename** :

```
#include <stdio.h>

int main() {
    const char* ancien_nom = "ancien_nom.txt";
    const char* nouveau_nom = "nouveau_nom.txt";

    if (rename(ancien_nom, nouveau_nom) == 0) {
        printf("Le fichier a été renommé avec succès.\n");
    } else {
        perror("Erreur lors du renommage du fichier");
        return 1;
    }

    return 0;
}
```

Revision #2
Created 2023-12-27 08:44:51 UTC by Elieroc
Updated 2023-12-27 08:59:26 UTC by Elieroc