

מטלה 7 במערכות הפעלה

להגשה עד ה-7.5.22 בשעה 00:00:00

מערכת קבצים בתוך קובץ

ממשו מערכת קרובה ל `ufs` בתוך קובץ. עליכם לממש את הפונקציות הבאות

`mymkfs(1)`

תוכנית היוצרת את הקובץ שבתוכו תמצא המערכת קבצים מקבלת פרמטר `-s` המתארת את גודל מערכת הקבצים. במערכת הקבצים שלכם יוקצו באופן אוטומטי 10% מהשטח לnode-ים בלוק אחד להיות סופר בלוק והשאר בלוקים רגילים. בנוסף ניתן להניח ש`mkfs` מייצרת את הספרייה הראשית `./`.

ממשו את הפונקציות הבאות ואת מבנה הנתונים `mydirent`

```
int mymount(const char *source, const char *target,
            const char *filesystemtype, unsigned long
            mountflags, const void *data);
int myopen(const char *pathname, int flags);
int myclose(int myfd);
ssize_t myread(int myfd, void *buf, size_t count);
ssize_t mywrite(int myfd, const void *buf, size_t count);
off_t mylseek(int myfd, off_t offset, int whence);
myDIR *myopendir(const char *name);
struct mydirent *myreaddir(myDIR *dirp);
int myclosedir(myDIR *dirp);
```

הקלות והנחיות נוספות

1. אין הגבלה על שדות ב* `myDIR`. מבנה `mydirent` חייב להכיל שדה `d_name` המכיל את שם הקובץ ויכול להכיל שדות נוספים כרצונכם.
2. עליכם לנהל טבלה של קבצים פתוחים `myfd` – תוכלו לשמור בטבלה הזאת מידה כרצונכם. אתם רשאים להניח שיש לכם משתנה גלובלי `struct myopenfile[MAX_FILES]` כאשר `MAX_FILES` קבוע וערכו 10000. אתם רשאים להכניס כל מידע שתרצו ל`struct myopenfile` ניתן להגדיר משתנים גלובלים נוספים אם אתם מוצאים בזה צורך.
3. `myopen` יעבוד רק על מערכת קבצים שנרכבה על ידי `mymount`.
4. `myread`, `mywrite`, `mylseek`, `myclose` יעבדו רק על קבצים שנפתחו על ידי `myopen`.
5. כל הפונקציות יהיו זמינות בעזרת ספרייה `libmyfs.so` וניתנות לשימוש על ידי `#include` אחד `myfs.h`.
6. לא צריך לתמוך ב `fragments` ובהרשאות.
7. אם תצטרכו לאתחל משתנים לפני שהספרייה שלכם עולה תוכלו להשתמש ב `((__attribute__((constructor))` מצרף מדריך בנושא זה [https://www.geeksforgeeks.org/attribute-__attribute__\(\(constructor\)\)-__attribute__\(\(destructor\)\)-syntax-c/](https://www.geeksforgeeks.org/attribute-__attribute__((constructor))-__attribute__((destructor))-syntax-c/)

מעל מערכת הקבצים (שמומשה בסעיף הקודם) שלכם עליכם לממש את המבנה

`myFILE *`

ואת הפונקציות

```
myFILE *myfopen(const char *restrict pathname, const char
*restrict mode)
int myfclose(myFILE *stream);
size_t myfread(void *restrict ptr, size_t size, size_t nmemb,
myFILE *restrict stream);
size_t myfwrite(const void *restrict ptr, size_t size, size_t
nmemb, myFILE *restrict stream);
int myfseek(myFILE *stream, long offset, int whence);
int myfscanf(myFILE *restrict stream,
const char *restrict format, ...);
int myfprintf(myFILE *restrict stream,
const char *restrict format, ...);
```

הפונקציות האלה יהיו בעלות פעולה ו prototype לפונקציות libc המתאימות. (לדוגמא פונקציית myfopen תהיה זהה לפונקציית fopen) הפונקציות שלכם ישתמשו במבנה `myFILE *` במקום `FILE *`.

הקלות והנחיות נוספות

1. אין הגבלה על שדות במבנה `myFILE *`.
2. אינכם נדרשים לתמוך ב `buffered I/O` ובנושא `flush` כאשר הפונקציות שלכם מבצעות קריאה או כתיבה פשוט בצעו זאת ללא המתנה
3. `myfopen` – צריך לתמוך במודים הבאים בלבד
 - a. `r` – קריאה
 - b. `r+` – קריאה וכתיבה
 - c. `w` – כתיבה (פתח את הקובץ לכתיבה ואם יש בו כבר תוכן מחק את התוכן הקיים)
 - d. `a` – `append` (צרף את התוכן החדש בסוף הקובץ הקיים)
4. `myprintf` ו `myscanf` צריכים לתמוך באפשרויות הבאות בלבד
 - a. `%d` – קריאה או כתיבה של `int`
 - b. `%c` – קריאה או כתיבה `char`
 - c. `%f` – קריאה או כתיבה של `float`
 - d. אינכם נדרשים לאפשרויות נוספות או לדיוק (`%2f`) (לדוגמא)
5. כל מבני הפונקציות צריכות להיות זמינות על ידי ספריה אחת `libmylibc.so` ועל ידי פעולות `include#` אחת ל `mystdio.h` header
6. בתרגיל זה אסור להשתמש בפונקציות `libc` (כלומר אסור לקרוא ל `fread` מתוך `myfread` וכדומה)
7. לצורך מימוש ב `fscanf` ו `fprintf` תצטרכו להשתמש ב `ellipsis` מצרף מדריך ל `ellipsis`
8. אין דרישה להשתמש ב `errno` (זה בונים) אפשר להחזיר -1 בכל הכשלונות.

<https://www.cprogramming.com/tutorial/c/lesson17.html>

בונוס (2 נקודות לציון הסופי)

1. ממש את errno עבור הפונקציות למעלה (1 נקודה לציון הסופי)
2. ממש את errno בעזרת thread local storage או thread specific storage עבור errno בפונקציות למעלה (1 נקודה נוספת לציון הסופי)
מצורף מדריך לשימוש ב-TLS
<https://www.ibm.com/docs/en/i/7.4?topic=techniques-data-that-is-private-thread>

הנחיות נוספות

1. גם תרגיל זה יבדק בעזרת MOSS
2. יש לצרף תוכנית בדיקה ודוגמאות הרצה (בעזרת script)
3. עליכם לצרף Makefile הבונה את כל הסעיפים ואת תוכניות הבדיקה.
4. תרגיל ה file system יכול לשמש אתכם גם בתרגיל 9 ו 10. ממליץ על מימוש יעיל ונוח.
5. חג שמח לכול מי שחוגג את יום העצמאות, עיד אל פיטר ואת עלייתה של מכבי יפו לליגה הלאומית!