



شرح پروژه

هدف این پروژه طراحی و پیاده‌سازی یک برنامه کاربردی multi-thread برای ادغام فایل‌های txt و سرهم کردن آن‌ها در قالب یک فایل txt کلی نحوه عملکرد برنامه شما به این شکل است که ابتدا فایل‌هایی را که در عمیق‌ترین مسیر نسبت به مسیر اصلی هستند را انتخاب می‌کند و در بین فایل‌های با یک عمق فایلی را که از همه قدیمی‌تر است در ابتدا به فایل اضافه می‌کنیم. (در ادامه صورت پروژه جزئیات و تمام خواسته‌ها ذکر شده است).

برنامه شما باید از مفاهیم بیان شده در درس مانند **فرآیندها، رشته‌ها و مکانیزم‌های همگام‌سازی** استفاده کند.
نکته: منظور از فایل‌های txt کلی فایلی است که شامل محتوای فایل‌های txt آن مسیر است.

مفاهیم پروژه

ایجاد فرآیند

در مسیر اولیه داده شده شما باید با یک فرآیند والد شروع کنید و پس از آن برای تمامی زیر پوشه‌ها، یک فرآیند جدید ایجاد کنید.

مدیریت فرآیندها

در هر پوشه یک فرآیند فرزند ایجاد شده و این فرآیند به ازای هر فایل یک thread ایجاد می‌کند، که به بررسی فایل‌های هر پوشه می‌پردازد و نتیجه هر پردازش یا عمل در آن پوشه توسط فرآیند فرزند در فایلی به فرمت "directory_name.log" ثبت می‌شود. (کدام ترد کدام فایل را خوانده است و کدام پروسس داخل کدام پوشه رفته است)

مکانیسم‌های همگام‌سازی

از قفل‌ها و سمافورها برای رسیدگی به مشکلات همگام‌سازی که ممکن است به دلیل داده‌های مشترک ایجاد شود، استفاده کنید.

ارتباطات بین‌فرآیندی

در این بخش شما باید مکانیسم‌های ارتباط بین‌فرآیندی (IPC) مانند لوله‌ها (Pipes)، صف‌های پیام (Message Queue) و حافظه مشترک (Shared Memory) را پیاده‌سازی کنید. با این کار می‌توان فرآیندهای فرزند را قادر ساخت تا داده‌های مورد نظر خود را به فرآیند والدی که آن‌ها را ایجاد کرده بازگردانند.

مواردی که برنامه‌ی شما باید بتواند انجام دهد

1. نمایش تعداد کل فایل‌های بررسی‌شده
2. ساخت یک فایل نهایی
3. ساخت یک فایل نهایی همراه توضیحات که هر بخش متن مختص به کدام فایل است کدام ترد آن را خوانده است و چه پروسسی آن را به والد خود داده است و غیره...
4. ایجاد یک فایل log در هر پوشه که نشان‌دهنده این است که در این پوشه کدام فایل‌ها بوده‌اند و کدام یک از آن‌ها اولویت بیشتری دارد. (در این فایل شما باید مشخص کنید که کدام thread، کدام process، فایل را بررسی کرده.)

محیط اجرای برنامه

برنامه برای اجرا نیاز به این دارد که کاربر مسیر پوشه اصلی را تعیین کند که بتواند عملیات لازم را انجام دهد به همین خاطر از terminal برای این کار استفاده خواهیم کرد.

نمره اضافه

رابط گرافیکی

اگر کاربر از هرگونه رابط گرافیکی جدیدی استفاده کند که دریافت مسیر و نمایش خروجی را در محیطی خارج از محیط terminal انجام دهد، شامل نمره اضافه خواهد بود.

راه‌های تشخیص فایل‌های تکراری

نمایش درختی برای نمایش نحوه مدیریت فرآیند‌ها و thread‌ها که نشان دهد هر مسیر شامل چه فایل‌هایی است.

توضیحات و فرمت ورودی و خروجی

کاربر پس از اجرای برنامه باید مسیر ورودی (مسیر پوشه اصلی) را از کاربر گرفته :

```
enter the directory of Path: /path/to/your/directory
```

تصویر 1: فرمت ورودی

و در ادامه برای مثال یک مورد را بررسی می‌کنیم اگر چینش فایل‌ها به شکل زیر باشد:

```
.
├── directory/
│   ├── dir1/
│   │   └── file1.txt
│   ├── dir2/
│   │   └── file2.txt
│   └── dir3/
│       ├── dir4/
│       │   └── dir5/
│       │       └── file3.txt
```

در هنگام ساخت فایل ابتدا محتویات file3 را قرار می‌دهیم چون در عمق بیشتری است و بین فایل‌های یک و دو چون هر دو در یک عمق قرار دهند فایل که قدیمی‌تر است ابتدا در فایل نوشته می‌شود بنابراین محتویات فایل نهایی به شکل زیر خواهد بود:

```
[$] cat finalfile.txt
file3
file2
file1
```

و یک فایل که مشخص کند کدام پروسس و کدام ترد فایل را بررسی کرده:

```

$ cat finalfilelog.txt
PID: 1004 DIR: dir3
PID: 1005 DIR: dir4
PID: 1006 DIR: dir5
TID: 100003
file3
Depth:3
modified: 8 Feb 17:05

PID: 1003 DIR: dir2
TID: 100002
file2
Depth:1
modified: 1 Feb 9:03

PID: 1002 DIR: dir1
TID: 100001
file1
Depth:1
modified: 1 Feb 10:02

```

توضیحات تکمیلی

- (1) پروژه به صورت انفرادی است.
- (2) زبان پیاده‌سازی پروژه، فقط زبان برنامه‌نویسی C می‌باشد. (برای پیاده‌سازی رابط گرافیکی، محدودیت زبانی ندارید.)
- (3) مراحل پیاده‌سازی و نحوه‌ی اجرای برنامه‌ی خود را حتماً در فایل **readme.md** به صورت کامل توضیح دهید.
- (4) فایل نهایی (شامل کد و فایل readme.md) را به فرمت **"concatenator<Student.IDs>_<Student.names>.zip"** در Vu بارگذاری کنید.
- (6) در صورت مشاهده هرگونه شباهت میان دانشجویان **نمره 100-** به هر دو نفر داده می‌شود.

مهلت تحویل: جمعه 26 اردیبهشت ماه 1404 خورشیدی؛ ساعت 23:59.

"موفق باشید"