

Sistemas Operacionais: Editor de Texto

...

Cristiano Lima - Ilem Santos

Implementar um simples editor de texto que execute a seguinte funcionalidade

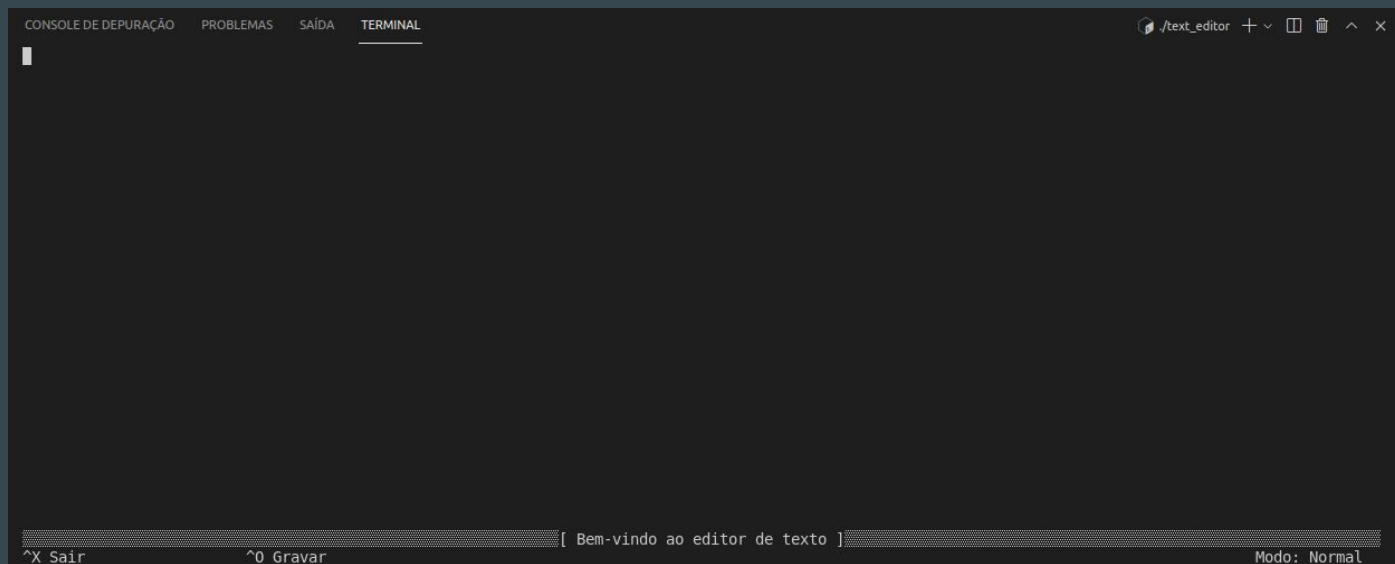
- Salvar somente números;
- Salvar somente letras;
- Salvar somente caracteres especiais;
- Salvar o número de linhas já escritos.

Utilize threads para identificar os itens a serem salvos e crie 4

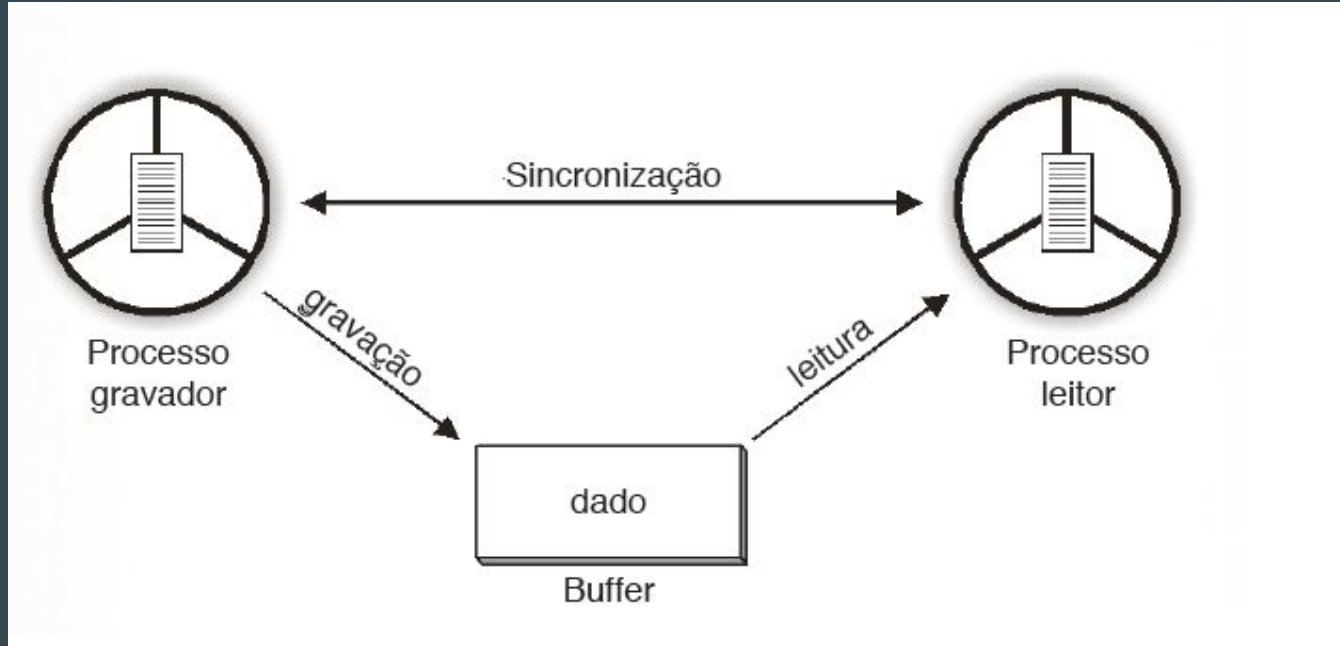
threads para alterar cada arquivo simultaneamente.

NCURSES

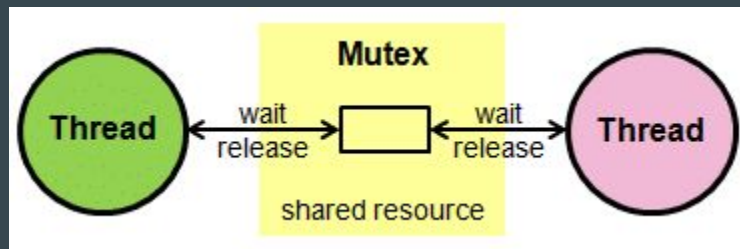
É uma biblioteca que provê uma API para o desenvolvimento de interfaces em modo texto.



Conflitos



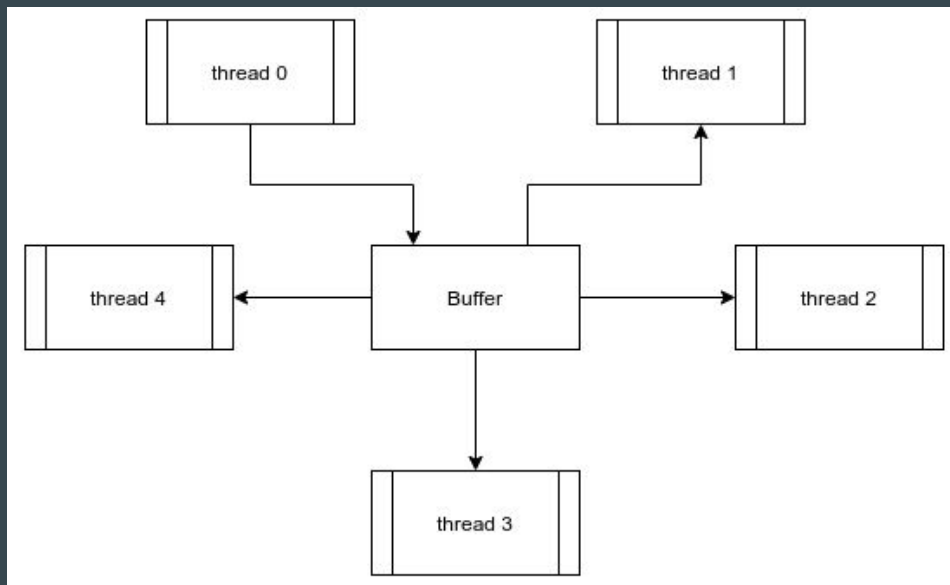
Exclusão mútua



Técnica utilizada em programação concorrente com a finalidade de evitar que dois processos tenham acesso simultaneamente a um recurso compartilhado, Chamada de seção crítica.

Semáforo

Variável utilizada para controlar o acesso a recursos compartilhados



```
typedef struct _rwlock_t {  
    sem_t lock_1;           /* trava binaria */  
    sem_t lock_2;           /* trava binaria */  
    sem_t wrt;              /* permite 1 escritor */  
    int readCount;          /* n de leitores */  
    pthread_mutex_t mutex;  
    pthread_cond_t cond_mutex;  
} rwlock_t;
```

```

int texteditor(texteditor_t *p) {
    while (!exitflag) {
        sem_wait(&p->ptr_semaphore->lock_2);
        sem_wait(&p->ptr_semaphore->wrt);

        /* [ Seção Crítica ] */

        sem_post(&p->ptr_semaphore->wrt);
        sem_post(&p->ptr_semaphore->lock_2);

        pthread_mutex_lock(&p->ptr_semaphore->mutex);
        pthread_cond_wait(&p->ptr_semaphore->cond_mutex, &p->ptr_semaphore->mutex);
        pthread_mutex_unlock(&p->ptr_semaphore->mutex);
    }

    return 0;
}

```

```

void my_thread_1(texteditor_t *rw) {
    while (state_condition) {
        sem_wait(&rw->ptr_semaphore->lock_2);
        sem_wait(&rw->ptr_semaphore->lock_1);
        rw->ptr_semaphore->readCount++;
        if (rw->ptr_semaphore->readCount == 1)
            sem_wait(&rw->ptr_semaphore->wrt);
        sem_post(&rw->ptr_semaphore->lock_1);
        sem_post(&rw->ptr_semaphore->lock_2);

        /* [ Seção Crítica ] */

        sem_wait(&rw->ptr_semaphore->lock_1);
        rw->ptr_semaphore->readCount--;
        if (rw->ptr_semaphore->readCount == 0)
            sem_post(&rw->ptr_semaphore->wrt);
        sem_post(&rw->ptr_semaphore->lock_1);
        pthread_cond_signal(&rw->ptr_semaphore->cond_mutex);
    }
}

```

Implementação

