**Варіант №3**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№**  **Варіанта** | **Передача даних** | **Каталоги** | **Ввод/Вивід** | **Системний визов ядра** | **Синхронування** |
| **3** | **Локальний потоковий сокет** | **Sockets/local** | **Текстові рядки (бібліотечні функції стандартні бібліотеки clibs)** | **Stdio (cистемний визов ядра)** | **Filelocks (файлове блокування)** |

**Задача №1. Розробити програму для копіювання рядків текстового файла у новий порожній файл.**

Алгоритм виконання програми повинен передбачати створення двох процесів (потоків), один з яких здійснює читання, передачу текстових рядків іншому процесу (потоку), призначеному для їх запису у цільовий файл. Кожен з процесів (потоків) оповіщає про результат наступної операції читання/запису процес (потік), який ініціює їх виконання, обробляє статус завершення та формує звіт. Передача даних між процесами (потоками) повинна бути організована через **локальний потоковий сокет**.

**Задача №2. Розробити програму для обміну вмісту двох текстових файлів між собою.**

Алгоритм виконання програми повинен передбачати створення двох процесів (потоків), кожен з яких здійснює читання блоків даних одного з файлів та їх запис у інший. Запис блоку даних у файл здійснюється при дозвільному стані сигнального об’єкта, який встановлюється процесом (потоком), що виконує читання блоку даних з файлу у проміжний буфер. Кожен з процесів (потоків) оповіщає про результат кожної операції читання/запису процес (потік), який ініціює їх виконання, обробляє статус завершення та формує звіт.

Порядок оперування сигнальним об’єктом, обумовлений вибором одного з можливих способів його створення:

-**файлове блокування.**

*Введення/виведення*. За умови дотримання коректності вирішення прикладної задачі згідно її постановки читання/запис текстових рядків (блоків даних) може бути здійснено із застосуванням однієї з можливих груп функцій програмного інтерфейсу введення/виведення зі складу стандартної бібліотеки мови програмування призначених для:

**Введення/виведення текстових рядків.**

*Спосіб оповіщення*. Постановкою задач обох типів передбачено оповіщення кожним процесом (потоком), який виконує операцію читання/запису про її результат процесу (потоку), який ініцював його виконання. За умови дотримання коректності вирішення задачі згідно її постановки пропонується один з наступних способів передачі таких відомостей:

**-сигнали реального часу.**

**Самостійна розробка програми**

Програма реалізації прикладної задачі згідно варіанта завдання повинна задовольняти наступним вимогам:

1. початковий текст програми розробляється мовою програмування С з наступною побудовою (компіляцією) файла програми, що виконується, засобами інтерфейсу командного рядка ОС на основі ядра Linux.
2. Код та дані програми, необхідні для виконання її алгоритму розміщуються (після після компіляції) в єдиному файлі, що виконується. Код процесів-нащадків (нових потоків) оформлюється у вигляді функцій.
3. Файл програми, що виконується, іменується **3-kostenko-2201.**
4. Запуск програми на виконання (демонстрація роботи програми) здійснюється з робочого каталогу, який розміщується в домашньому каталозі, з використанням командного інтерпретатора bash у операційному середовищі ОС на основі ядра Linux.
5. Командний рядок для запуску програми на виконання: ./prog file1 file2, де prog-імя файла програми, що виконується: **3-kostenko-2201,** file1, file2-імена файлів, які є об’єктами для виконання операцій читання/запису згідно умов постановки прикладних задач.
6. При спробі запуску програми на виконання без опцій (file1 та/або file2) або без визначення їх необхідної кількості здійснюється виведення стислої довідки про порядок використання програми та завершення роботи.
7. Обробка помилок виконання функцій програмного інтерфейсу початкового коду програми з наданням відповідних повідомлень користувачу.
8. Завершення програми з наданням відповідного повідомлення користувачу при виникненні аварійних ситуацій, які впливають на хід виконання алгоритму виконання прикладної задачі.
9. Виведення звіту про результат роботи програми перед її завершенням, який містить: час початку роботи; кількість прочитаних рядків (блоків); кількість записаних рядків (блоків); час завершання роботи.

**Робота повинна бути в текстовому вигляді з поясненями та в електронному програма (файли) для запуску їх в Linux.**

Додаю каталоги, які надані були для зразків для відпрацювання контрольної.