БГУИР

Кафедра ЭВМ

Отчет по лабораторной работе № 5

Тема: «Потоки исполнения, взаимодействие и синхронизация»

Выполнил:

студент группы 150502 Альхимович Н.Г.

Проверил:

старший преподаватель Поденок Л.П.

Минск

2023

**1 ЦЕЛЬ РАБОТЫ**

Реализовать задачу «производители-потребители» с использованием потоков в рамках одного процесса.

**2 ПОСТАНОВКА ЗАДАЧИ**

Основной процесс создает очередь сообщений, после чего ожидает и обрабатывает нажатия клавиш, создавая и закрывая потоки двух типов – производители и потребители, а также изменяя размер очереди. Очередь сообщений представляет собой классическую структуру — кольцевой буфер, содержащий указатели на сообщения, и пара указателей на голову и хвост. Помимо этого, очередь содержит счетчик добавленных сообщений и счетчик извлеченных.

Производители формируют сообщения и, если в очереди есть место, перемещают их туда.

Потребители, если в очереди есть сообщения, извлекают их оттуда, обрабатывают и освобождают память с ними связанную.

Для работы используются два семафора для заполнения и извлечения, а также мьютекс или одноместный семафор для монопольного доступа к очереди.

Формат сообщений представлен на рисунке 2.1.

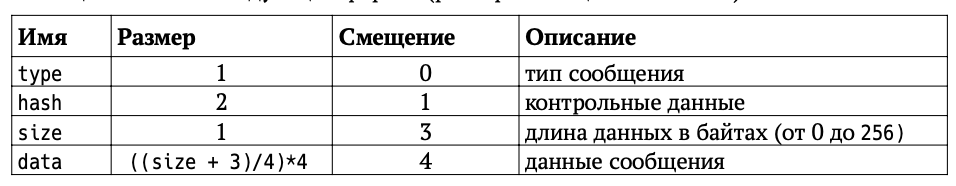


Рисунок 2.1 – Формат сообщений

Производители генерируют сообщения, используя системный генератор rand(3) для size и data. В качестве результата для size используется остаток от деления на 257. Если остаток от деления равен нулю, rand(3) вызывается повторно. Если остаток от деления равен 256, значение size устанавливается равным 0, реальная длина сообщения при этом составляет 256 байт.

При формировании сообщения контрольные данные формируются из всех байт сообщения. Значение поля hash при вычислении контрольных данных принимается равным нулю. Для расчета контрольных данных можно использовать любой подходящий алгоритм на выбор студента.

После помещения значения в очередь перед освобождением мьютекса очереди производитель инкрементирует счетчик добавленных сообщений. Затем после поднятия семафора выводит строку на stdout, содержащую новое значение этого счетчика.

Потребитель, получив доступ к очереди, извлекает сообщение и удаляет его из очереди. Перед освобождением мьютекса очереди инкрементирует счетчик извлеченных сообщений. Затем после поднятия семафора проверяет контрольные данные и выводит строку на stdout, содержащую новое значение счетчика извлеченных сообщений.

Следует предусмотреть задержки, чтобы вывод можно было успеть прочитать в процессе работы программы.

Следует предусмотреть защиту от тупиковых ситуаций из-за отсутствия производителей или потребителей.

**3 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ**

Результат работы программы с опцией c:

nina@nina-MacBook-AIr:~/Документы/Альхимович Н.Г./Lab05:main -$ make

gcc -W -Wall -Wextra -Wno-unused-parameter -Wno-unused-variable -std=c11 -pedantic main.c -o main

nina@nina-MacBook-AIr:~/Документы/Альхимович Н.Г./Lab05:main -$ ./main

Выберите действие:

c - создать поток-производитель;

+ - создать поток-потребитель;

d - закрыть поток-производитель;

- - закрыть поток-потребитель;

> - увеличить размер очереди;

< - уменьшить размер очереди;

q - завершить работу.

c

Поток-производитель 281469755257120 создан

Результат работы программы с опцией +:

+

Поток-потребитель 281469675106592 создан

Результат работы программы с опцией d:

d

Поток 281469755257120 закрыт

Осталось 0 потоков

Результат работы программы с опцией -:

-

Поток 281469675106592 закрыт

Осталось 0 потоков

Результат работы программы с опцией q:

q

Поток 281469755257120 закрыт

Все потоки закрыты

Потоки не были созданы

Вывод функции-производителя:

29-ое сообщение добавлено в очередь

Контрольные данные: 616

Длина данных: 4

Данные сообщения: 237 98 183 98

Вывод функции-потребителя:

20 сообщений извлечено из очереди

**4 ВЫВОД**

В ходе лабораторной работы была разработана программа, организующая создание потоков двух видов в зависимости от выбранной опции (производитель (producer), формирующий сообщение и добавляющий его в очередь, и потребитель (consumer), извлекающий сообщение из очереди), управляющий их работой и создающий очередь сообщений заданной структуры.