

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет информатики и
Радиоэлектроники»

Кафедра электронных вычислительных машин

Лабораторная работа №4
«Задача производителя-потребителя для процессов»

Выполнил:
студент группы 150501
Ковальчук Д.И.

Проверил:
старший преподаватель
_____ Поденок Л.П.

Минск
2023

1 ПОСТАНОВКА ЗАДАЧИ

Задача производители-потребители для процессов. Основной процесс создает очередь сообщений, после чего ожидает и обрабатывает нажатия клавиш, порождая и завершая процессы двух типов —производители и потребители.

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.

Сообщения имеют следующий формат (размер и смещение в байтах):

Имя	Размер	Смещение	Описание
type	1	0	тип сообщения
hash	2	1	контрольные данные
size	1	3	длина данных в байтах (от 0 до 256)
data	$((size + 3)/4)*4$	4	данные сообщения

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 помимо всего 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 помимо всего новое значение счетчика извлеченных сообщений.

При получении сигнала о завершении процесс должен завершить свой цикл и только после этого завершиться, не входя в новый. 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.

2 АЛГОРИТМ

- 1) Начало.
- 2) Выделение разделяемой памяти.
- 3) Присоединение разделяемой памяти к процессу.
- 4) Загрузка кольцевого буфера в разделяемую память.
- 5) Инициализация семафоров для producer и consumer.
- 6) Ввод опции.
- 7) Опция "1", - создание процесса producer, переход на шаг 5.
- 8) Опция "2" – завершение процесса producer, переход на шаг 5.
- 9) Опция "3" – создание процесса consumer, переход на шаг 5.
- 10) Опция "4" – завершение процесса consumer, переход на шаг 5.
- 11) Опция "l" – вывод информации о всех процессах.
- 12) Опция "q" – завершение всех процессов (producer и consumer).
- 13) Отсоединение разделяемой памяти.
- 14) Освобождение разделяемой памяти.
- 15) Конец.

3 ТЕСТ

```
darya@darya-VivoBook-ASUSLaptop-X521EQ-S533EQ:~/osisp/lab04$ ./main
```

```
      MENU
1  CREATE PRODUCER
2  DELETE PRODUCER
3  CREATE CONSUMER
4  DELETE CONSUMER
m  SHOW MENU
l  SHOW PROCESSES INFO
q  EXIT PROGRAMM
1
PRODUCER CREATED. PID: 3656
PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 637F
SIZE: 66
AMOUNT OF PUSHED MESSAGES: 1
1
PRODUCER CREATED. PID: 3657
PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 0A3C
SIZE: 188
AMOUNT OF PUSHED MESSAGES: 2

PROCESS (PID): 3657 CREATED MESSAGE:
TYPE: 0
HASH: 637F
SIZE: 66
AMOUNT OF PUSHED MESSAGES: 3

PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 133D
```

SIZE: 170
AMOUNT OF PUSHED MESSAGES: 4
2
PROCESS (PID): 3657 CREATED MESSAGE:
TYPE: 0
HASH: 0A3C
SIZE: 188
AMOUNT OF PUSHED MESSAGES: 5

PRODUCER DELETED. PID: 3657
PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: FD4A
SIZE: 83
AMOUNT OF PUSHED MESSAGES: 6
3
CONSUMER CREATED. PID: 3658
PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: FAE6
SIZE: 58
AMOUNT OF PUSHED MESSAGES: 7

PROCESS (PID): 3658 CONSUMED MESSAGE:
TYPE: 0
HASH: 637F
SIZE: 66
AMOUNT OF EXTRACTED MESSAGES: 1

PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 865C
SIZE: 132
AMOUNT OF PUSHED MESSAGES: 8

PROCESS (PID): 3658 CONSUMED MESSAGE:
TYPE: 0
HASH: 0A3C
SIZE: 188
AMOUNT OF EXTRACTED MESSAGES: 2
4
PROCESS (PID): 3658 CONSUMED MESSAGE:
TYPE: 0
HASH: 637F
SIZE: 66
AMOUNT OF EXTRACTED MESSAGES: 3

CONSUMER DELETED. PID: 3658
PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: E7E1
SIZE: 92
AMOUNT OF PUSHED MESSAGES: 9

PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 1F17
SIZE: 222
AMOUNT OF PUSHED MESSAGES: 10

PROCESSES(PID)
MAIN
3655
PRODUCERS
3656

CONSUMERS

PROCESS (PID): 3656 CREATED MESSAGE:
TYPE: 0
HASH: 6EEA
SIZE: 123
AMOUNT OF PUSHED MESSAGES: 11

4 ВЫВОД

В ходе лабораторной работы была решена задача производителяпотребители для процессов, были изучены механизмы работы общей памяти, семафоров, были предусмотрены тупиковые ситуации, если отсутствуют потребители или производители. Также были использованы такие функции как `mmap()`, `ftruncate()`, `munmap()`, `shm_open()`, `shm_unlink()`, `sem_open()`, `close()`, `sem_unlink()`, `sem_wait()`, `sem_post()`.