

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии

Микропроект № 2

по дисциплине

"Архитектура вычислительных систем"

Пояснительная записка

Исполнитель:

студент группы БПИ192

Гураевский М. Д.

Москва 2020

Содержание

Введение	3
Наименование программы	3
Текст задания	3
Документы, на основании которых ведётся разработка	3
Описание программы	3
Тестирование	4
Текст программы	6
Формат вывода	7
Список литературы	8

Введение

Наименование программы

Данная программа представляет собой упрощённую версию задачи о писателях и читателях – регулируются только действия читателей.

Текст задания

Задача о читателях и писателях-2 («грязное чтение»). Базу данных разделяют два типа потоков – читатели и писатели. Читатели выполняют транзакции, которые просматривают записи базы данных, транзакции писателей и просматривают, и изменяют записи. Предполагается, что в начале БД находится в непротиворечивом состоянии (т. е. отношения между данными имеют смысл). Транзакции выполняются в режиме «грязного чтения», то есть процесс-писатель не может получить доступ к БД только в том случае, если ее занял другой процесс-писатель, а процессы-читатели ему не мешают. Создать многопоточное приложение с потоками-писателями и потоками-читателями. Реализовать решение, используя семафоры, и не используя блокировки чтения-записи.

Документы, на основании которых ведётся разработка

Программа выполнена в рамках задания по дисциплине "Архитектура вычислительных систем", в соответствии с учебным планом подготовки бакалавров по направлению 09.03.04 «Программная инженерия», 2 курс 2 модуль.

Основанием для разработки является письмо профессора факультета компьютерных наук Легалова Александра Ивановича от 28.11.20

Описание программы

Базу данных разделяют два типа потоков – читатели и писатели. Читатели выполняют транзакции, которые просматривают записи базы данных, транзакции писателей и просматривают, и изменяют записи. Предполагается, что в начале БД находится в непротиворечивом состоянии (т. е. отношения между данными имеют смысл). Транзакции выполняются в режиме «грязного чтения», то есть процесс-писатель не может получить доступ к БД только в том случае, если ее занял другой процесс-писатель, а процессы-читатели ему не мешают. Создать многопоточное приложение с потоками-писателями и потоками-читателями. Реализовать решение, используя семафоры, и не используя блокировки чтения-записи

Тестирование

```
Operation 21 || Writer 2: reads old value 1 and writes new value 18 to the cell [8]
Operation 41 || Writer 1: reads old value 0 and writes new value 5 to the cell [0]
Operation 41 // Reader 1: reads value 12 from cell [10]
Operation 42 || Writer 1: reads old value 5 and writes new value 7 to the cell [2]
Operation 42 // Reader 1: reads value 15 from cell [9]
Operation 5 // Reader 9: reads value 7 from cell [2]
Operation 15 // Reader 3: reads value 18 from cell [5]
Operation 8 // Reader 6: reads value 0 from cell [6]
Operation 15 || Writer 3: reads old value 10 and writes new value 12 to the cell [7]
Operation 22 // Reader 2: reads value 12 from cell [10]
Operation 22 || Writer 2: reads old value 0 and writes new value 11 to the cell [4]
Operation 43 || Writer 1: reads old value 12 and writes new value 10 to the cell [10]
Operation 43 // Reader 1: reads value 7 from cell [2]
Operation 44 // Reader 1: reads value 16 from cell [3]
Operation 44 || Writer 1: reads old value 15 and writes new value 7 to the cell [9]
Operation 9 // Reader 5: reads value 18 from cell [5]
Operation 5 // Reader 10: reads value 12 from cell [7]
Operation 9 || Writer 5: reads old value 12 and writes new value 11 to the cell [7]
Operation 12 // Reader 4: reads value 5 from cell [0]
Operation 12 || Writer 4: reads old value 7 and writes new value 19 to the cell [2]
Operation 23 // Reader 2: reads value 18 from cell [5]
Operation 23 || Writer 2: reads old value 18 and writes new value 4 to the cell [5]
Operation 45 // Reader 1: reads value 10 from cell [10]
Operation 45 || Writer 1: reads old value 16 and writes new value 9 to the cell [3]
Operation 7 // Reader 7: reads value 19 from cell [2]
Operation 16 // Reader 3: reads value 19 from cell [2]
Operation 16 || Writer 3: reads old value 5 and writes new value 9 to the cell [0]
Operation 46 // Reader 1: reads value 2 from cell [1]
Operation 46 || Writer 1: reads old value 4 and writes new value 14 to the cell [5]
```

(10 читателей, 5 писателей, по 100 операций у каждого)

```
Operation 0 // Reader 1: reads value 4 from cell [4]
Operation 0 // Reader 3: reads value 3 from cell [3]
Operation 0 // Reader 2: reads value 2 from cell [2]
Operation 0 || Writer 1: reads old value 8 and writes new value 0 to the cell [8]
Operation 1 // Reader 1: reads value 10 from cell [10]
Operation 1 || Writer 1: reads old value 0 and writes new value 15 to the cell [0]
Operation 2 // Reader 1: reads value 9 from cell [9]
Operation 1 // Reader 2: reads value 9 from cell [9]
Operation 2 || Writer 1: reads old value 1 and writes new value 3 to the cell [1]
Operation 1 // Reader 3: reads value 15 from cell [0]
Operation 3 // Reader 1: reads value 15 from cell [0]
Operation 3 || Writer 1: reads old value 0 and writes new value 8 to the cell [8]
Operation 2 // Reader 2: reads value 7 from cell [7]
Operation 4 || Writer 1: reads old value 4 and writes new value 5 to the cell [4]
Operation 4 // Reader 1: reads value 6 from cell [6]
Operation 5 // Reader 1: reads value 2 from cell [2]
Operation 5 || Writer 1: reads old value 2 and writes new value 7 to the cell [2]
Operation 2 // Reader 3: reads value 7 from cell [2]
Operation 3 // Reader 2: reads value 10 from cell [10]
Operation 6 || Writer 1: reads old value 3 and writes new value 14 to the cell [1]
Operation 6 // Reader 1: reads value 3 from cell [3]
Operation 7 || Writer 1: reads old value 10 and writes new value 12 to the cell [10]
Operation 7 // Reader 1: reads value 6 from cell [6]
Operation 4 // Reader 2: reads value 12 from cell [10]
Operation 8 || Writer 1: reads old value 12 and writes new value 9 to the cell [10]
Operation 8 // Reader 1: reads value 7 from cell [2]
Operation 3 // Reader 3: reads value 7 from cell [7]
Operation 9 // Reader 1: reads value 5 from cell [5]
Operation 9 || Writer 1: reads old value 7 and writes new value 16 to the cell [2]
Operation 5 // Reader 2: reads value 15 from cell [0]
```

(3 читателя, 1 писатель, по 10 операций у каждого)

```

Operation 0 // Reader 83: reads value 7      from cell [7]
Operation 0 // Reader 84: reads value 7      from cell [7]
Operation 0 // Reader 94: reads value 1      from cell [1]
Operation 0 // Reader 78: reads value 7      from cell [7]
Operation 0 // Reader 96: reads value 7      from cell [7]
Operation 0 // Reader 88: reads value 9      from cell [9]
Operation 0 // Reader 89: reads value 0      from cell [0]
Operation 0 // Reader 90: reads value 1      from cell [1]
Operation 0 // Reader 82: reads value 4      from cell [4]
Operation 0 // Reader 92: reads value 4      from cell [4]
Operation 0 // Reader 93: reads value 9      from cell [9]
Operation 0 // Reader 76: reads value 7      from cell [7]
Operation 0 // Reader 95: reads value 6      from cell [6]
Operation 0 // Reader 87: reads value 2      from cell [2]
Operation 0 // Reader 97: reads value 10     from cell [10]
Operation 0 // Reader 98: reads value 1      from cell [1]
Operation 0 // Reader 99: reads value 3      from cell [3]
Operation 0 // Reader 100: reads value 8     from cell [8]
Operation 0 || Writer 1: reads old value 3   and writes new value 19 to the cell [3]
Operation 1 || Writer 1: reads old value 0   and writes new value 17 to the cell [0]
Operation 2 || Writer 1: reads old value 9   and writes new value 18 to the cell [9]
Operation 3 || Writer 1: reads old value 8   and writes new value 3 to the cell [8]
Operation 4 || Writer 1: reads old value 4   and writes new value 7 to the cell [4]
Operation 5 || Writer 1: reads old value 5   and writes new value 14 to the cell [5]
Operation 6 || Writer 1: reads old value 7   and writes new value 12 to the cell [4]
Operation 7 || Writer 1: reads old value 6   and writes new value 10 to the cell [6]
Operation 8 || Writer 1: reads old value 14  and writes new value 5 to the cell [5]
Operation 9 || Writer 1: reads old value 7   and writes new value 5 to the cell [7]
Operation 10 || Writer 1: reads old value 10 and writes new value 5 to the cell [10]
Operation 11 || Writer 1: reads old value 3  and writes new value 3 to the cell [8]

```

(100 читателей, 1 операция, 1 читатель, 100 операций)

```

Operation 1 || Writer 90: reads old value 1   and writes new value 2 to the cell [3]
Operation 20 // Reader 4: reads value 6      from cell [0]
Operation 20 || Writer 4: reads old value 7   and writes new value 10 to the cell [10]
Operation 40 // Reader 2: reads value 8      from cell [5]
Operation 40 || Writer 2: reads old value 6   and writes new value 15 to the cell [0]
Operation 80 || Writer 1: reads old value 14  and writes new value 16 to the cell [9]
Operation 80 // Reader 1: reads value 16     from cell [9]
Operation 1 // Reader 91: reads value 15     from cell [0]
Operation 1 || Writer 91: reads old value 15  and writes new value 13 to the cell [6]
Operation 8 // Reader 11: reads value 10     from cell [10]
Operation 12 // Reader 7: reads value 4      from cell [2]
Operation 8 || Writer 11: reads old value 16  and writes new value 3 to the cell [8]
Operation 12 || Writer 7: reads old value 16  and writes new value 10 to the cell [9]
Operation 81 || Writer 1: reads old value 13  and writes new value 10 to the cell [4]
Operation 81 // Reader 1: reads value 4      from cell [2]
Operation 4 // Reader 23: reads value 10     from cell [4]
Operation 1 // Reader 92: reads value 10     from cell [9]
Operation 2 // Reader 46: reads value 5      from cell [7]
Operation 4 || Writer 23: reads old value 10  and writes new value 4 to the cell [4]
Operation 2 || Writer 46: reads old value 8   and writes new value 2 to the cell [5]
Operation 1 || Writer 92: reads old value 5   and writes new value 15 to the cell [7]
Operation 7 // Reader 13: reads value 4      from cell [4]
Operation 7 || Writer 13: reads old value 13  and writes new value 18 to the cell [6]
Operation 14 // Reader 6: reads value 15     from cell [7]
Operation 14 || Writer 6: reads old value 15  and writes new value 4 to the cell [7]
Operation 28 // Reader 3: reads value 2      from cell [3]
Operation 41 // Reader 2: reads value 4      from cell [4]
Operation 41 || Writer 2: reads old value 10  and writes new value 16 to the cell [9]
Operation 28 || Writer 3: reads old value 2   and writes new value 17 to the cell [5]
Operation 82 || Writer 1: reads old value 4   and writes new value 19 to the cell [7]
Operation 82 // Reader 1: reads value 2      from cell [3]
Operation 3 // Reader 31: reads value 4      from cell [2]
Operation 1 // Reader 93: reads value 16     from cell [9]
Operation 3 || Writer 31: reads old value 4   and writes new value 18 to the cell [2]
Operation 1 || Writer 93: reads old value 18  and writes new value 17 to the cell [2]
Operation 83 || Writer 1: reads old value 17  and writes new value 7 to the cell [5]
Operation 83 // Reader 1: reads value 15     from cell [0]
Operation 2 // Reader 47: reads value 15     from cell [0]
Operation 1 // Reader 94: reads value 17     from cell [2]

```

100 писателей, 100 читателей, 100 операций у каждого

Текст программы

```
#define HAVE_STRUCT_TIMESPEC
#define _CRT_SECURE_NO_WARNINGS
#define _CRT_RAND_S
#define RAND_MAX
#include <stdio.h>
#include <semaphore.h>
#include <thread>
#include <iostream>
#include <random>

using namespace std;

mt19937 mersenne(static_cast<unsigned int>(time(0)));
int database[11] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
sem_t semaphore;

void dataWriter(int threadNum) {
    int data;
    int id;
    int result;

    for (int i = 0; i < 100; i++) {
        sem_wait(&semaphore);

        data = mersenne() % 20;
        id = mersenne() % 11;
        result = database[id];
        database[id] = data;
        printf("Operation %d || Writer %d: reads old value %d \tand writes new value %d to the\n",
            i, threadNum, result, data, id, result);

        sem_post(&semaphore);
        this_thread::sleep_for(chrono::seconds(threadNum));
    }
}

void dataReader(int threadNum) {
    int result;
    int id;

    for (int i = 0; i < 100; i++) {
        id = mersenne() % 11;
        result = database[id];
        printf("Operation %d // Reader %d: reads value %d \tfrom cell [%d]\n", i, threadNum,
            result, id);
        this_thread::sleep_for(chrono::seconds(threadNum));
    }
}

int main() {
    const int countReader = 100;
    const int countWriter = 100;

    sem_init(&semaphore, 0, 1);

    thread writers[countReader];
    thread readers[countReader];

    for (int i = 0; i < countReader; i++)
        readers[i] = thread(dataReader, i + 1);

    for (int i = 0; i < countWriter; i++)
        writers[i] = thread(dataWriter, i + 1);

    for (int i = 0; i < countWriter; ++i)
        writers[i].join();

    for (int i = 0; i < countReader; ++i)
        readers[i].join();

    sem_destroy(&semaphore);
    return 0;
}
```

Формат вывода

В начале строки выводится номер операции у данного типа и номера потока, далее определяется тип потока (“Writer” или “Reader”), а также его порядковый номер. После, в зависимости от типа, строка продолжается либо что и откуда он прочитал (это в случае, если тип потока “Reader”), либо что и откуда прочитал, а после на что заменил (это в случае, если тип потока “Writer”). Порядковый номер писателя или читателя – это номер потока, под которым он лежит.

Опишу по отдельности:

Строка потока *читателя*

1. Operation {Номер операции}
2. // (Выводится для более удобного форматирования строк зрительно)
3. Reader {Номер потока в массиве потоков типа Reader}
4. reads value {Значение читаемое из ячейки}
5. from cell [{индекс в БД данной ячейки, из который читатель читал}]

Строка потока *писателя*

1. Operation {Номер операции}
2. || (Выводится для более удобного форматирования строк зрительно)
3. Writer {Номер потока в массиве потоков типа Writer}
4. reads old value {Значение читаемое из ячейки}
5. and writes new value {Новое задаваемое значение в ячейку БД}
6. to the cell [{индекс в БД данной ячейки, из который читатель читал}]

Список литературы

1. https://www.youtube.com/watch?v=4GdTcqE0iqg&feature=emb_logo&ab_channel=davidko
2. <https://ravesli.com/urok-71-generatsiya-sluchajnyh-chisel-funktsii-srand-i-rand/#toc-5>
3. https://learnc.info/c/pthreads_semaphores.html
4. <ftp://sourceware.org/pub/pthreads-win32>