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

Факультет компьютерных наук  
Департамент программной инженерии

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

ЗАДАЧА О ЧИТАТЕЛЯХ И ПИСАТЕЛЯХ.

Вариант 3.

Автор пояснительной записки, \_\_\_\_\_\_\_\_\_\_\_\_ Доржсурэн Тоголдор

студент ФКН,

“Программная инженерия”,

группа БПИ193

Москва 2020

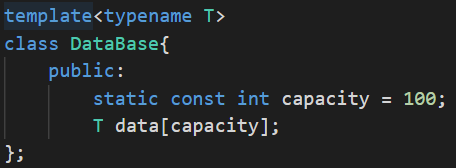
# Постановка задачи

Текст задания: Базу данных разделяют два типа процессов – 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). Каждая отдельная транзакция переводит БД из одного непротиворечивого состояния в другое. Для предотвращения взаимного влияния транзакций процесс-писатель должен иметь исключительный доступ к БД. Если к БД не обращается ни один из процессов-писателей, то выполнять транзакции могут одновременно сколько угодно читателей. Создать многопоточное приложение с потоками-писателями и потоками-читателями. Реализовать решение, используя семафоры.

# Алгоритм решения задачи

Для решения как, было указано в задаче использовались семафоры для разделения общей БД между потоками-писателей и потоками-читателей с исключительным доступом для писателей.

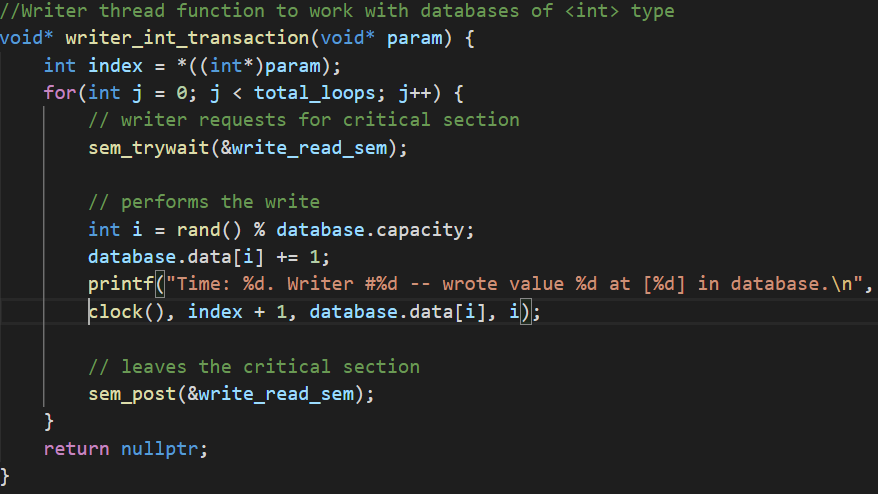
База данных в рамках задачи реализована классом <Database> с параметром <T> для типа объекта в контейнере.



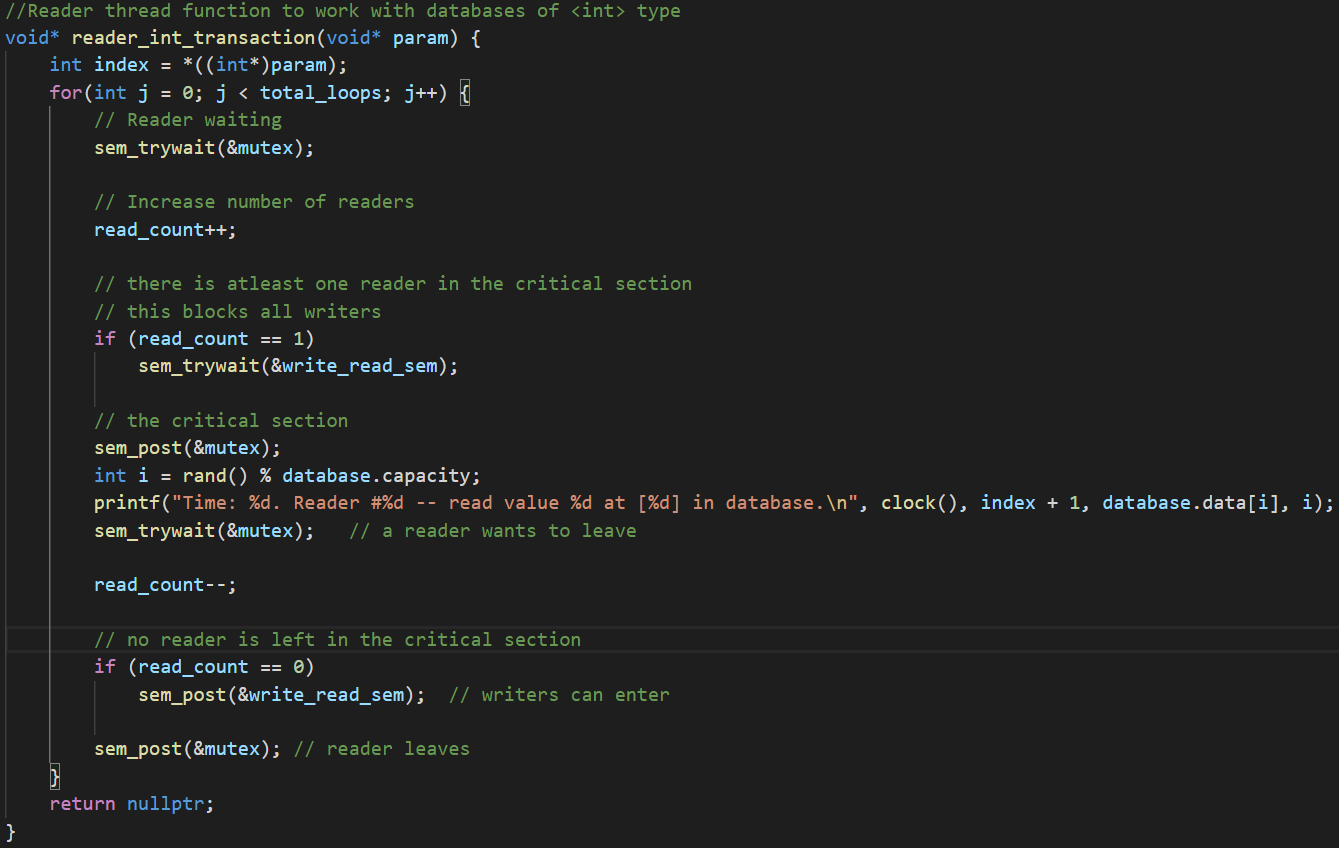
В задаче используются два общих семафора:



Метод потоков-писателей использует семафор <write\_read\_sem> для критической секции:



Метод потоков-читателей использует оба семафора <write\_read\_sem> и <mutex> для критической секции и для уведомления потоков-писателей, проверяет наличие активных читателей с помощью счётчика <read\_count>:



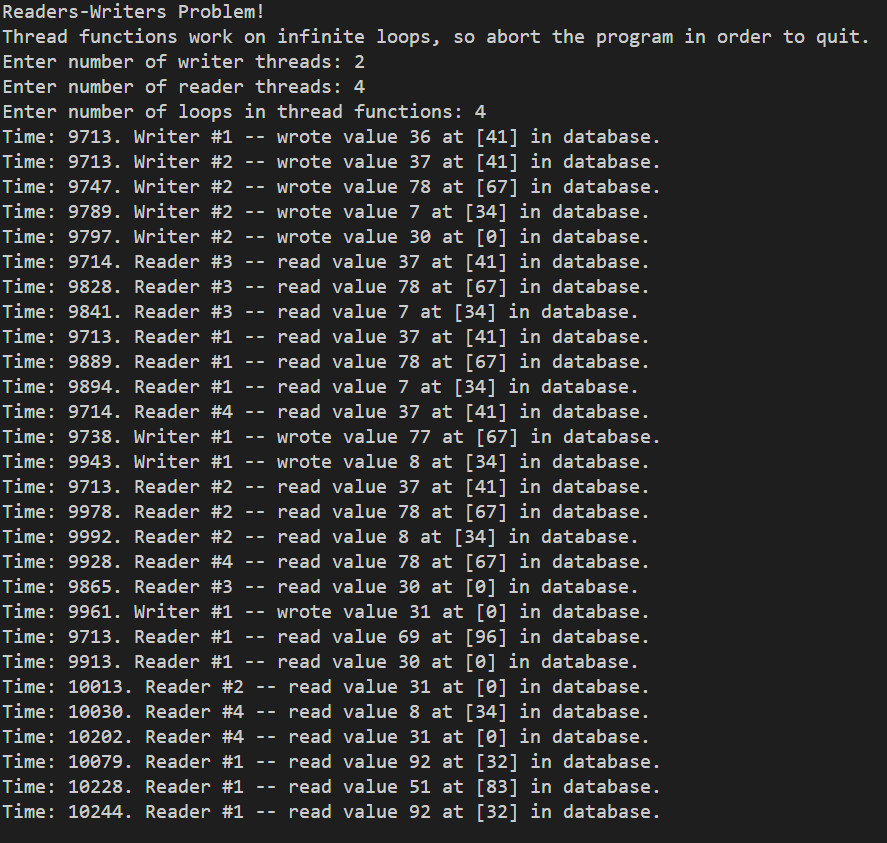
# Описание области допустимых значений входных параметров

Программа при запуске запрашивает у пользователя 3 входных параметра:

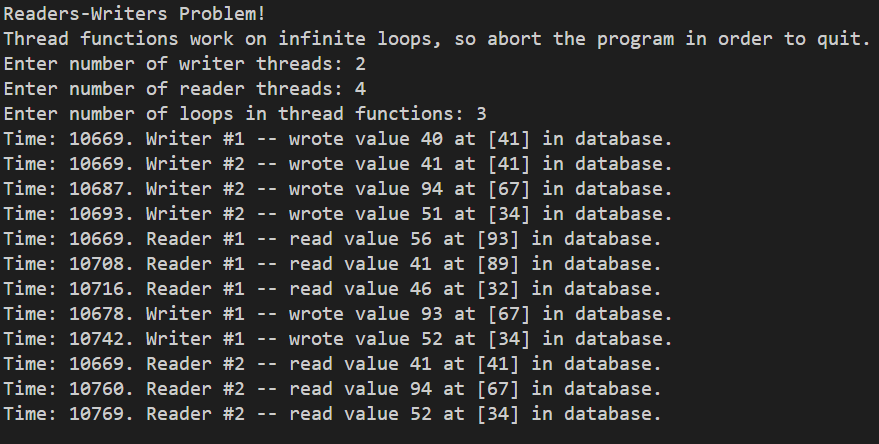
* количество потоков-писателей (1 < целое число < 5).
* количество потоков-читателей (1 < целое число < 5).
* количество циклов в методах потоков (1 < целое число < 500).

# примеры Работы программы

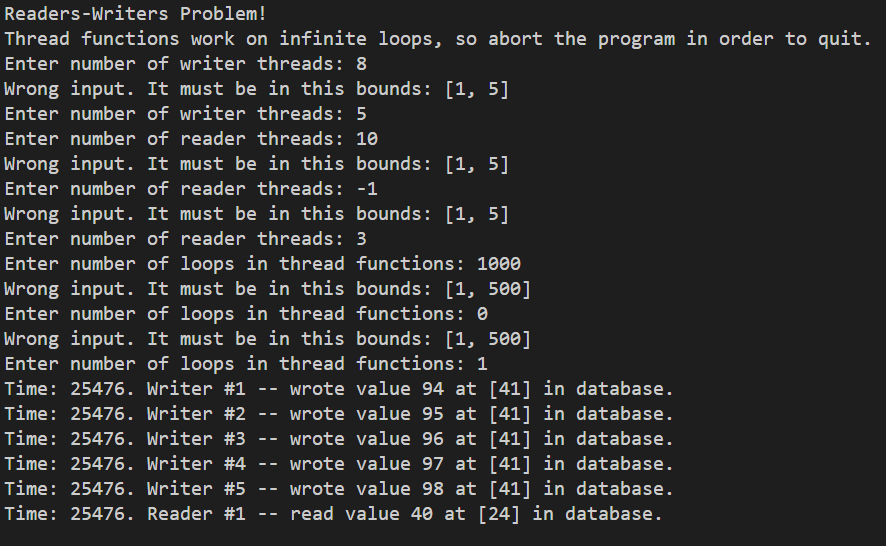
**Пример 1**

****

**Пример 2**

****

**Пример 3**

****

# Список источников

1. Семафоры, онлайн ресурс. Доступ: <https://learnc.info/c/pthreads_semaphores.html>
2. Producer–consumer problem, Доступ: <https://en.wikipedia.org/wiki/Producer%E2%80%93consumer_problem>
3. Многопоточное программирование. Синхронизация, онлайн ресурс. Доступ: <http://softcraft.ru/edu/comparch/practice/thread/02-sync/>