Задача 1.

Создать класс на языке C#, который:

- называется TaskQueue и реализует логику пула потоков;

- создает указанное количество потоков пула в конструкторе;

- содержит очередь задач в виде делегатов без параметров:

delegate void TaskDelegate();

- обеспечивает постановку в очередь и последующее выполнение делегатов с помощью метода

void EnqueueTask(TaskDelegate task);

Задача 2.

Создать класс на языке C#, который:

- называется Mutex и реализует двоичный семафор с помощью атомарной операции Interlocked.CompareExchange.

- обеспечивает блокировку и разблокировку двоичного семафора с помощью public-методов Lock и Unlock.

Задача 3.

Создать на языке C# класс ActionRunner, реализующий метод RunAndWaitAll, который:

- принимает в параметрах массив делегатов;

- выполняет все указанные делегаты параллельно с помощью пула потоков;

- дожидается окончания выполнения всех делегатов.

Привести простейший пример использования созданного класса.

Задача 4.

Создать класс на языке C#, который:

- называется NativeBuffer и обеспечивает выделение и освобождение блока памяти ОС через функции Marshal.AllocHGlobal и Marshal.FreeHGlobal.

- содержит свойство Handle, позволяющее получить дескриптор блока памяти ОС;

- реализует метод Finalize для автоматического освобождения блока памяти;

- реализует интерфейс IDisposable для принудительного освобождения блока памяти;

Задача 5.

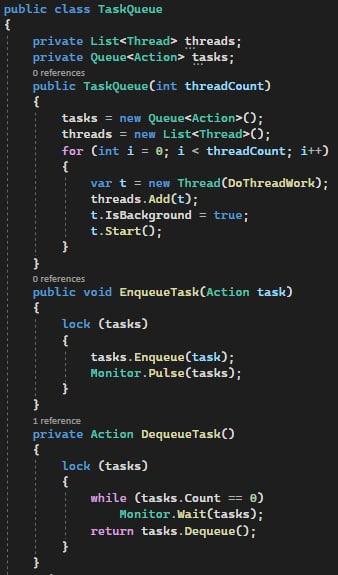
Реализовать консольную программу на языке C#, которая:

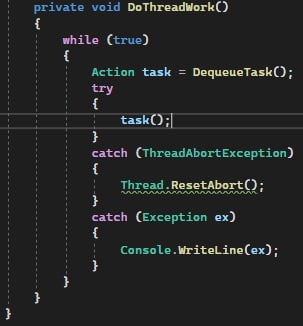
- принимает в параметре командной строки путь к сборке .NET (EXE- или DLL-файлу);

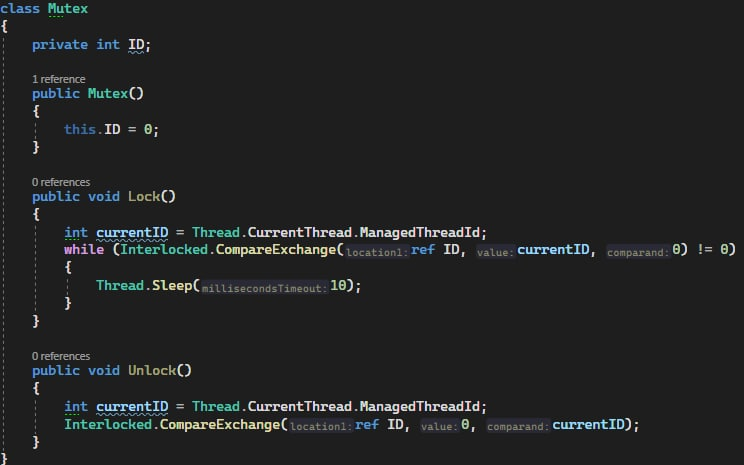
- загружает указанную сборку в память;

- выводит на экран полные имена всех public-типов данных этой сборки, упорядоченные по пространству имен (namespace) и по имени.

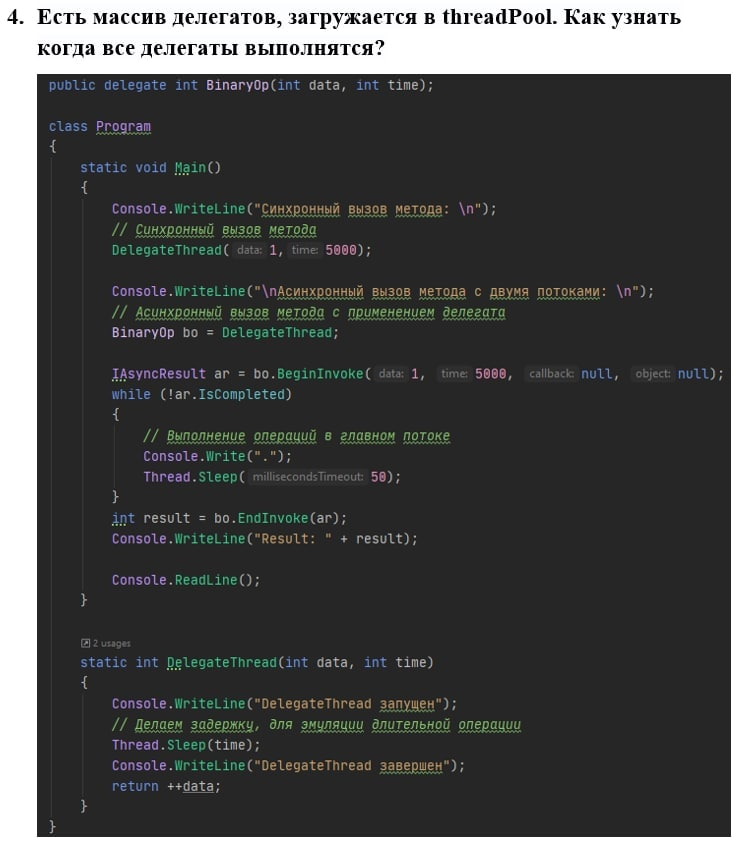
1 задача.



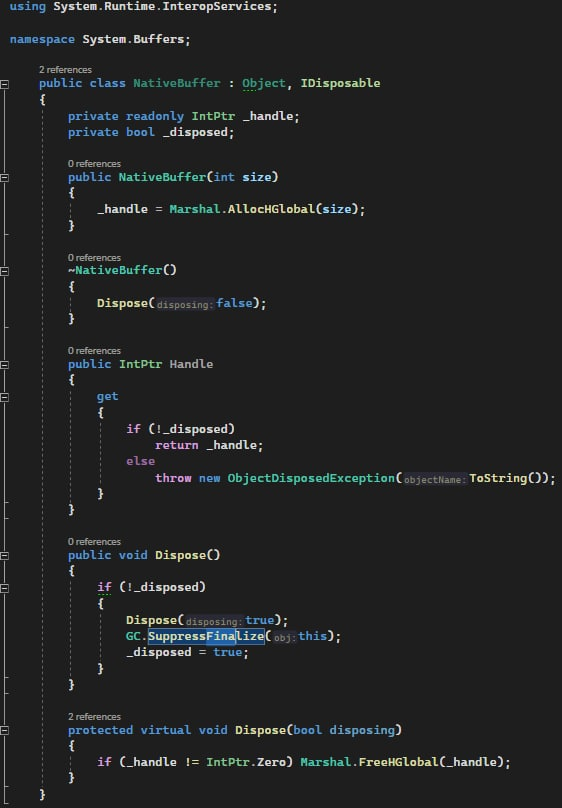


2 задача.   


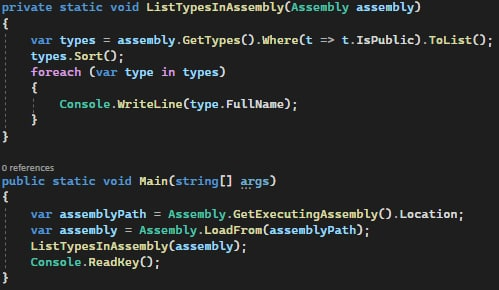
3 задача.



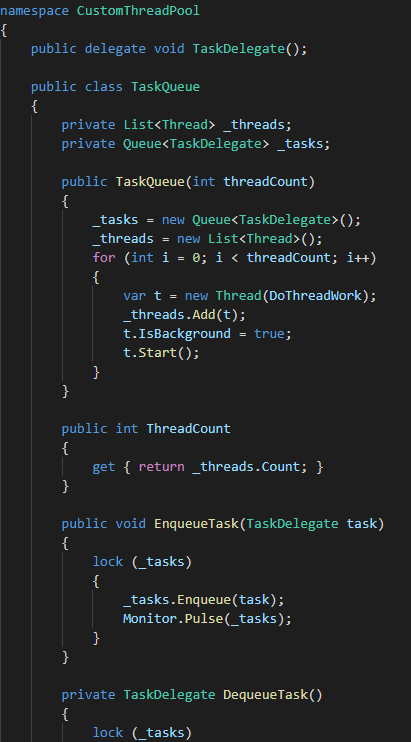
4 задача.

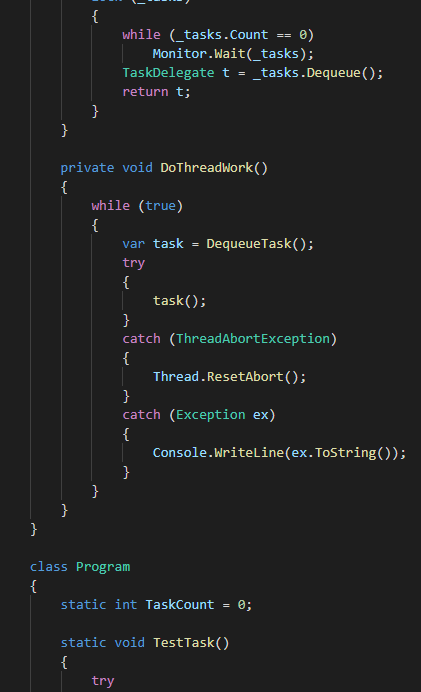


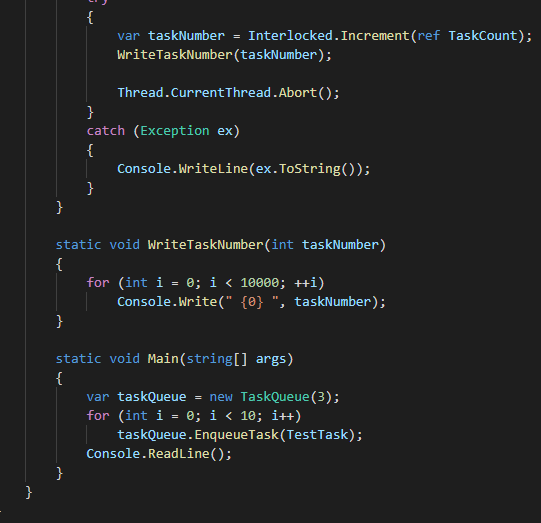
5 задача.



1 задача.

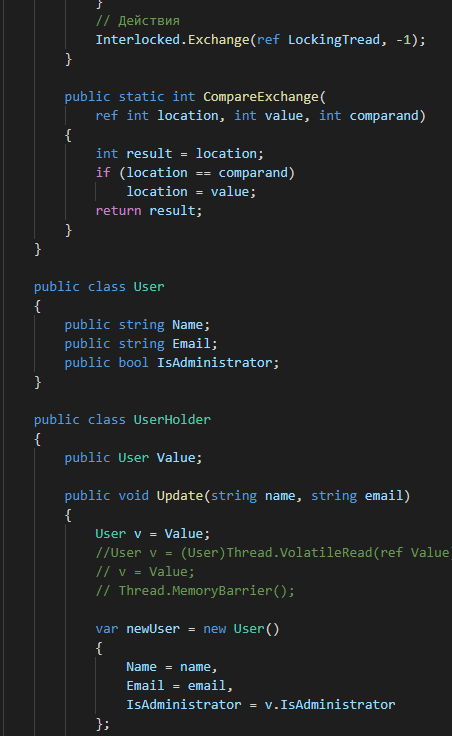


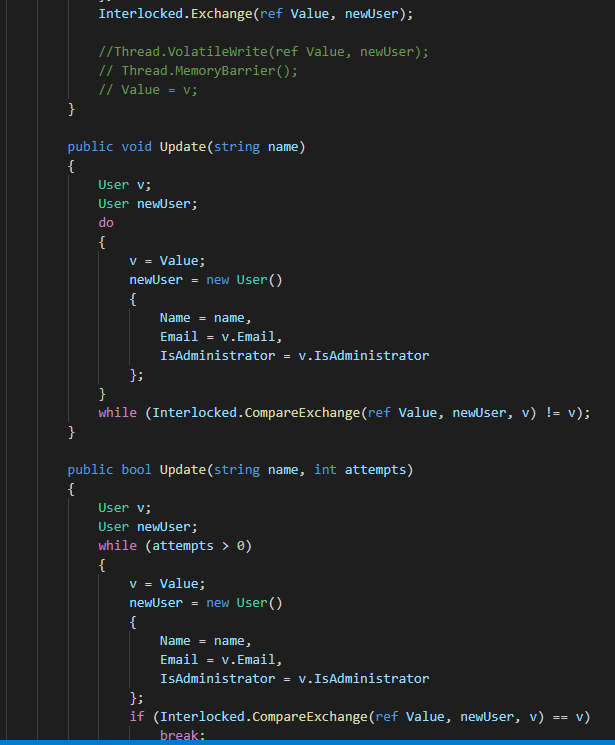


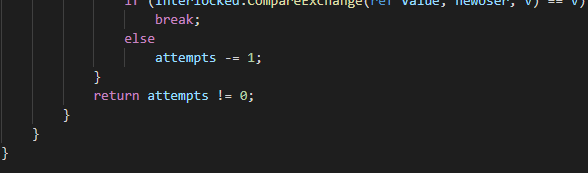


2 задача.









4 задача.



