using System.Threading.Channels;

internal class Program

{

static void Main(string[] args)

{

//3

Semaphore semaphore = new Semaphore(3,3);

Thread[] thread =

{

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

new Thread(() =>

{

semaphore.WaitOne();

Random random = new Random();

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} Random:{random.Next(1,100)}");

semaphore.Release();

}),

};

foreach(Thread t in thread)

t.Start();

foreach (Thread t in thread)

t.Join();

//2

/\* int[] mass = {1,2,3,4,5 };

Mutex mutex = new Mutex();

Thread thread1 = new Thread(() =>

{

mutex.WaitOne();

Random random = new Random();

for (int i = 0; i < mass.Length; i++)

{

mass[i] += random.Next(1,11);

}

mutex.ReleaseMutex();

});

Thread thread2 = new Thread(() => {

mutex.WaitOne();

mass.ToList().ForEach(x => Console.Write(x+" "));

Console.WriteLine();

Console.WriteLine($"MAX: {mass.Max(x => x)}");

mutex.ReleaseMutex();

});

thread1.Start();

thread2.Start();

thread1.Join();

thread2.Join();\*/

//1

/\*Mutex mutex = new Mutex();

Thread thread = new Thread(

() =>

{

mutex.WaitOne();

for (int i = 0; i < 20; i++)

{

Console.WriteLine(i);

}

mutex.ReleaseMutex();

});

Thread thread1 = new Thread(

() =>

{

mutex.WaitOne();

for (int i = 10; i < 0; i--)

{

Console.WriteLine(i);

}

mutex.ReleaseMutex();

});

thread.Start();

thread1.Start();

thread.Join();

thread1.Join();\*/

}

}