

Krzysztof Pancierz
Programowanie współbieżne i rozproszone
Synchronizacja wątków za pomocą bariery

```
import java.util.concurrent.CyclicBarrier;
public class Glowna
{
    public static void main(String[] args)
    {
        CyclicBarrier bariera=new CyclicBarrier(3, new Watek2());

        new Watek1(bariera, "A").start();
        new Watek1(bariera, "B").start();
        new Watek1(bariera, "C").start();
    }
}

*****

import java.util.concurrent.BrokenBarrierException;
import java.util.concurrent.CyclicBarrier;
public class Watek1 extends Thread
{
    private CyclicBarrier bariera;
    private String nazwa;

    public Watek1(CyclicBarrier bariera, String nazwa)
    {
        this.bariera=bariera;
        this.nazwa=nazwa;
    }

    public void run()
    {
        for(int i=0; i<20; i++)
        {
            System.out.println("Wątek 1 o nazwie "+this.nazwa);
        }

        try
        {
            this.bariera.await();
        }
        catch(BrokenBarrierException e)
        {}
        catch(InterruptedException e)
        {}
    }
}

*****

public class Watek2 extends Thread
{
    public void run()
    {
        for(int i=0; i<20; i++)
        {
            System.out.println("Wątek 2");
        }
    }
}
```