練習問題　解答①

問題１

デッドロックは発生しない。

このプログラムには複数のスレッドがあるが、それらは1つのオブジェクトへのアクセスを待機するだけなのでロックを互いに保持することはない。

また、synchronizedメソッドq1( ) でsynchronizedメソッドq2( ) を呼び出しているが、オブジェクトのロックを取得する場合、スレッドは同じオブジェクトのsynchronizedメソッドならいくつでも呼び出すことができる。

```
class Q{
	synchronized void q1(){
		q2();
	}
	synchronized void q2(){
	}
}

class ThreadQ extends Thread {
	Q q;

	ThreadQ(Q q) {
		this.q = q;
	}

	public void run(){
		for(int i=0;i<100000;i++)
			q.q1();
	}
}
```

```
class DeadlockQ {
	private final static int NUMTHREADS = 10;
	public static void main(String[] args) {
		// オブジェクトの作成
		Q q = new Q();

		// スレッドの作成
		ThreadQ threads[] = new ThreadQ[NUMTHREADS];
		for(int i=0;i<NUMTHREADS;i++){
			threads[i] = new ThreadQ(q);
			threads[i].start();
		}

		// スレッドの完了を待つ
		for (int i=0;i<NUMTHREADS;i++)
			try{
				threads[i].join();
			}
			catch(Exception e){
				e.printStackTrace();
			}

		//メッセージの表示
		System.out.println("Done!");
	}
}
```

練習問題　解答②

```
class Mouse extends Thread {
  Box box;

  Mouse(Box box) {
    this.box = box;
  }

  public void run() {
    try {
      while(true) {
        // 箱の外で10以上20未満の秒間過ごす
        long out = (long)(10000 * Math.random() + 10000);
        Thread.sleep(out);

        // 箱に入る、もしくは入ろうとする
        box.enter(this);

        // 箱の中に5以上8未満の秒間留まる
        long in = (long)(3000 * Math.random() + 5000);
        Thread.sleep(in);

        // 箱から出る
        box.exit(this);
      }
    }
    catch(Exception e) {
      e.printStackTrace();
    }
  }
}
```

```
class Box {
  private static int CAPACITY = 4;
  int count;
  synchronized void enter(Mouse mouse) {
    // 箱がいっぱいのうちは待機する
    try {
      while(count == CAPACITY) {
        wait();
      }
    }
    catch(InterruptedException ie) {
      ie.printStackTrace();
      System.exit(0);
    }
    // カウンターを1つ増やす
    ++count;
    // カウンターを表示する
    System.out.println(count);
  }
  synchronized void exit(Mouse mouse) {
    // カウンターを1つ減らす
    --count;
    // カウンターを表示する
    System.out.println(count);
    // 待機中のスレッドに通知を送る
    notify();
  }
}
```

```
class BoxMouse {
  public static void main(String args[]) {
    Box box = new Box();

    for(int i = 0; i < 10; i++)
      new Mouse(box).start();
  }
}
```