

## **Races and locks**

---

How many data races does the following two-thread program have?

```
int counter = 0;
```

---

thread t

```
int cnt;
```

```
1 cnt = counter;
2 counter = cnt + 1;
```

thread u

```
int cnt;
```

```
cnt = counter;           1
cnt = cnt + 1;          2
System.out.println(cnt); 3
```

1. One
2. Two
3. None
4. It depends on the run

How many data races does the following two-thread program have?

```
int counter = 0;
```

---

thread t

```
int cnt;
```

```
1 cnt = counter;  
2 counter = cnt + 1;
```

thread u

```
int cnt;
```

```
cnt = counter;           1  
cnt = cnt + 1;          2  
System.out.println(cnt); 3
```

1. One
2. Two
3. None
4. It depends on the run

What does the following two-thread program print?

```
int counter = 0;
```

---

thread t

```
int cnt;
```

```
1 cnt = counter;  
2 counter = cnt + 1;
```

thread u

```
int cnt;
```

```
cnt = counter;           1  
cnt = cnt + 1;          2  
System.out.println(cnt); 3
```

1. It always prints 0
2. It always prints 1
3. It sometimes prints 0 and sometimes prints 1
4. It sometimes prints 1 and sometimes prints 2

What does the following two-thread program print?

```
int counter = 0;
```

---

thread t

```
int cnt;
```

```
1 cnt = counter;  
2 counter = cnt + 1;
```

thread u

```
int cnt;
```

```
cnt = counter;           1  
cnt = cnt + 1;          2  
System.out.println(cnt); 3
```

1. It always prints 0
2. It always prints 1
3. It sometimes prints 0 and sometimes prints 1
4. It sometimes prints 1 and sometimes prints 2

What are the possible values of counter upon termination?

```
int counter = 0;
```

---

thread t

```
int cnt;
1 for(i=0;i<10000;++i){
2   cnt = counter;
3   counter = cnt + 1;
4 }
```

thread u

```
int cnt;
1 for(i=0;i<10000;++i){
2   cnt = counter;
3   counter = cnt + 1;
4 }
```

1. It always prints 10000
2. It always prints 20000
3. It prints some value between 10000 and 20000
4. It prints some value between 2 and 20000

What are the possible values of counter upon termination?

```
int counter = 0;
```

---

thread t

```
int cnt;
1 for(i=0;i<10000;++i){
2   cnt = counter;
3   counter = cnt + 1;
4 }
```

thread u

```
int cnt;
1 for(i=0;i<10000;++i){
2   cnt = counter;
3   counter = cnt + 1;
4 }
```

1. It always prints 10000
2. It always prints 20000
3. It prints some value between 10000 and 20000
4. It prints some value between 2 and 20000

What does the following two-thread program print?

```
Lock one = new Lock(); Lock two = new Lock();
```

---

thread t

```
1 one.lock();
2 two.lock();
3 System.out.println("t");
4 two.unlock();
5 one.unlock();
```

thread u

```
1 one.lock();
2 two.lock();
3 System.out.println("u");
4 two.unlock();
5 one.unlock();
```

1. It prints "t" followed by "u".
2. It prints "u" followed by "t".
3. Either of the answers above.
4. Either of the answers above or it does not print anything.

What does the following two-thread program print?

```
Lock one = new Lock(); Lock two = new Lock();
```

---

	thread t		thread u	
1	one.lock();		one.lock();	1
2	two.lock();		two.lock();	2
3	System.out.println("t");		System.out.println("u");	3
4	two.unlock();		two.unlock();	4
5	one.unlock();		one.unlock();	5

1. It prints "t" followed by "u".
2. It prints "u" followed by "t".
3. **Either of the answers above.**
4. Either of the answers above or it does not print anything.

What does the following two-thread program print?

```
Lock one = new Lock(); Lock two = new Lock();
```

---

thread t

```
1 one.lock();
2 two.lock();
3 System.out.println("t");
4 two.unlock();
5 one.unlock();
```

thread u

```
1 two.lock();
2 one.lock();
3 System.out.println("u");
4 two.unlock();
5 one.unlock();
```

1. It prints "t" followed by "u".
2. It prints "u" followed by "t".
3. Either of the answers above.
4. Either of the answers above or it does not print anything.

What does the following two-thread program print?

```
Lock one = new Lock(); Lock two = new Lock();
```

---

thread t

```
1 one.lock();
2 two.lock();
3 System.out.println("t");
4 two.unlock();
5 one.unlock();
```

thread u

```
1 two.lock();
2 one.lock();
3 System.out.println("u");
4 two.unlock();
5 one.unlock();
```

1. It prints "t" followed by "u".
2. It prints "u" followed by "t".
3. Either of the answers above.
4. **Either of the answers above or it does not print anything.**