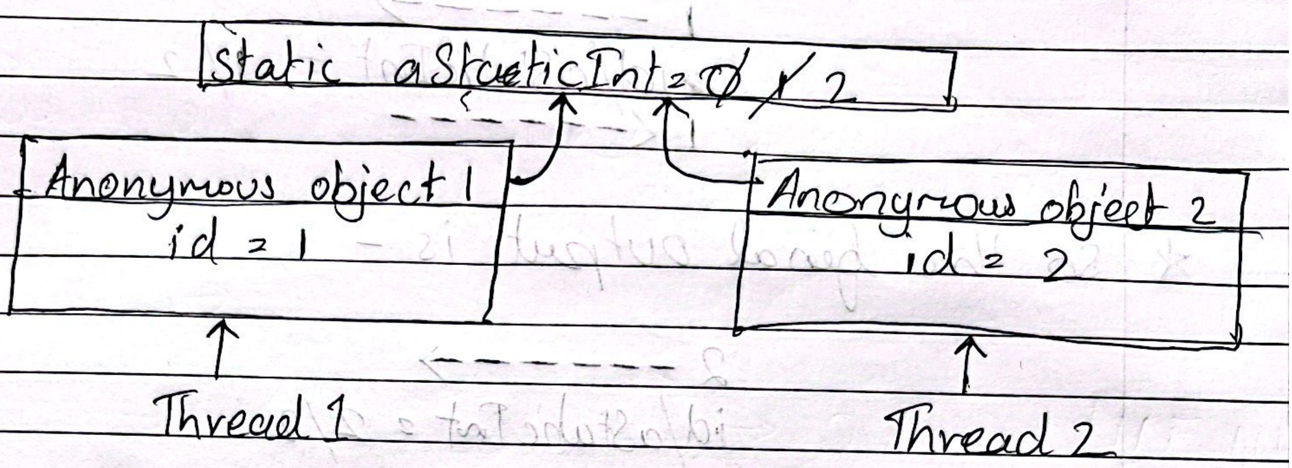
Sriverkatesh. Nair (sn6711)
Abhinav Menon (am6176)

Homework 11.2



①

→ In line 18, we ~~first~~ create the ~~first~~ anonymous object 1 i.e. "Thread 1".

→ We go into the constructor where we initialize id = 1 and aStaticInt = 1.

→ Since id = 1, therefore, we create anonymous object 2 i.e "Thread 2".

→ We again go into the constructor again and initialize id = 2 and aStaticInt = 2.

→ In line 10, we have called the run() method on the "Thread 2" so ~~we~~ the scheduler first completes the execution of run method for "Thread 2"

→ Within the run method -

2 ------->
id/aStaticInt = 2/2
2 <-------

→ After the complete execution of run method, we execute the start() method on "Thread 1".

→ Since ~~the~~ ~~no~~ there are no other threads that need to be executed within the main thread, the scheduler will execute the start method on "Thread 1" completely.

→ Within the run method for "Thread 1" -

```
1 ------->
id/aStaticInt = 1/2
1 <-------
```

★ So the final output is -

```
2 ------->
id/aStaticInt = 2/2
2 <-------
1 ------->
id/aStaticInt = 1/2
1 <-------
```

② Possible Output~~s~~?

```
1 ------->
id/aStaticInt = 1/2
2 ------->
id/aStaticInt = 2/2
2 <-------
1 <-------
```

No, this cannot be a possible output of this program because this output suggests that "Thread 1" starts executing before the complete creation of anonymous object 1 which cannot be the case. The scheduler ~~first~~ always first executes the run() method of "Thread 2" because the method call is done within the constructor of "Thread 1". After the complete creation of anonymous object 1, start() method is executed on "Thread1"