CSE12 - Lecture 23

Wednesday, May 31, 2023 8:00 AM

PA8 / PA5 Late / Resubmit → due today

PA6 Late / Resubmit → due Tuesday

No new PAs!

PA7 & PA8 Late/Resubmit → due Friday of Week 10 (no slip day)

Create a RandomStream class that generates random numbers in an enhanced for loop:.

Generating a random number:

From java.util.*

class Random

public int nextInt(int bound)

Returns a pseudorandom, uniformly distributed int value between 0 (inclusive) and the specified value (exclusive), drawn from this random number generator's sequence.

Random random; = new Random();
int value = random.nextInt(100); //random number between 0 and 99

Note: always use a field, never create a new one over and over again in a loop.

```
RandomStream r = new RandomStream(10, 100);
for (Integer i : r) {          while (itr.hasNext()) {
   System.out.println(i);        i = itr.next();
 }
                               }

public class RandomStream ~~(int size, int bound)~~ implements Iterable<Integer> {
   int size;
   int bound;
   Random random;
   public RandomStream(int size, int bound) {
       this.size = size;
       this.bound = bound;
       random = new Random( );     seed / debugging
   }
   class RandomIterator implements Iterator<Integer> {
       int current = 0;
       public boolean hasNext() {
           return this.current < size;
       }
       public Integer next() {
           int value = random.nextInt(bound);
           this.current += 1;
           return value;
       }
   }
   public Iterator<Integer> iterator() {
```

55

75

61

0 → start at 0

1

99 → bound - 1

7

10

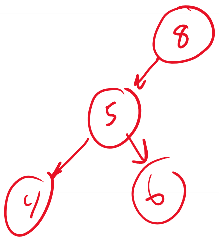
11

30

```
      }
      public Iterator<Integer> iterator() {
          return new RandomIterator();
      }
  }
```

} Name: ______________________ PID: ______________________ Code: 3851

How would we make a BST iterator?



post

pre

in-order traversal

Iterator class

- ↳ save state
  - ↳ create an array list
  - ↳ fill AL w/ an in-order traversal
- ↳ iterator → next/hasNext same as ArrayList

How would we make a Heap iterator?

① copy the heap contents into the Iterator (N)

- ↳ use the poll() of the copy in next() (lg(n))
  - ↳ all elements in heap order

② copy the array (contents) into the iterator (N)

- ↳ sort in heap order → N·log(N)
- ↳ in next() → use the array
  like an array list version of iterator