CSE12 - Lecture 22

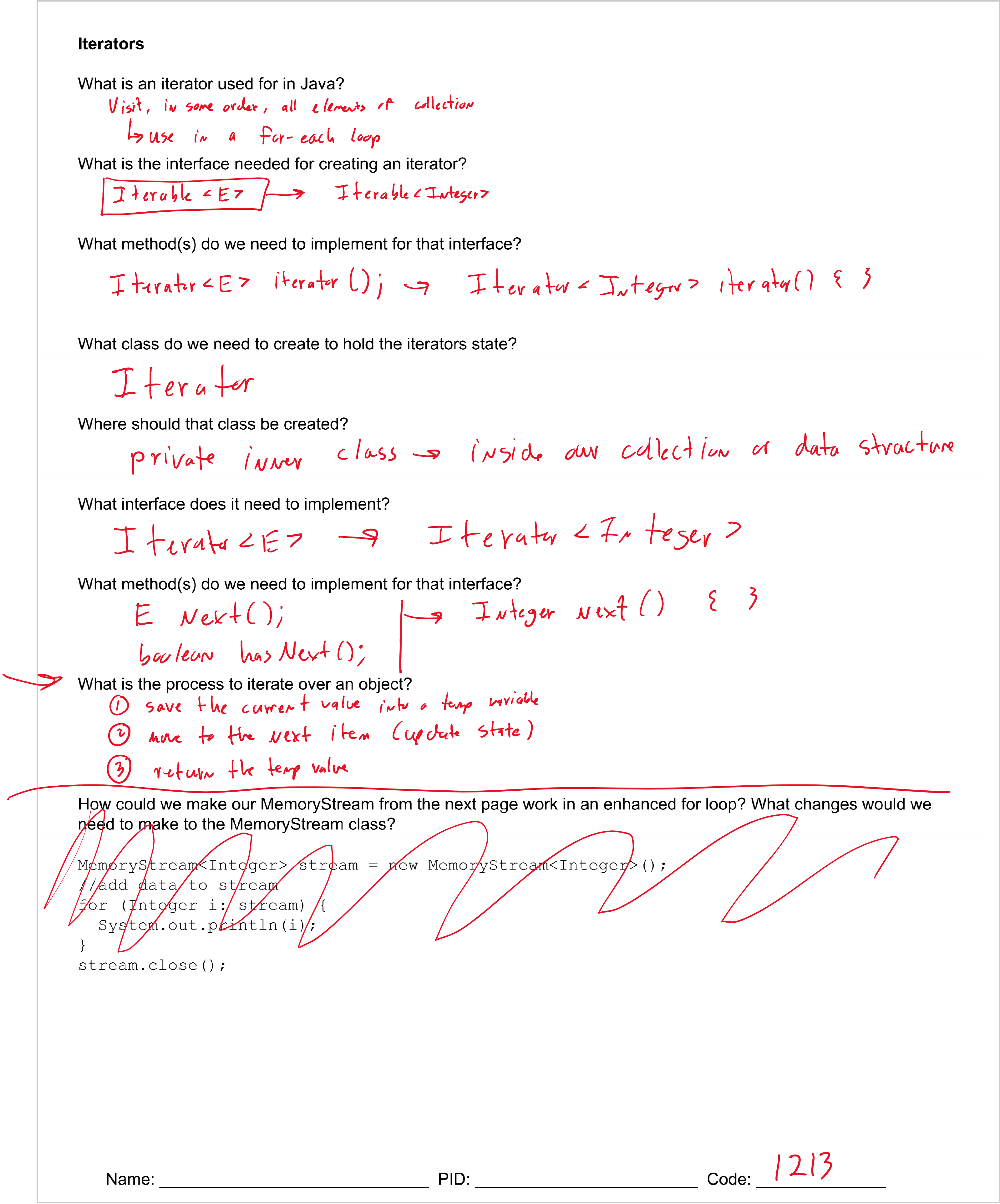
Wednesday, May 24, 2023 8:00 AM

Exam 2 → Friday → counting → hash tables

PA8 released → due Tuesday

PA4 Late / Resubmit / PA7 due today

Monday → no class → holiday



Iterators

What is an iterator used for in Java?

Visit, in some order, all elements of collection

↳ use in a for-each loop

What is the interface needed for creating an iterator?

$\boxed{\text{Iterable<E>}}$ → Iterable<Integer>

What method(s) do we need to implement for that interface?

Iterator<E> iterator(); → Iterator<Integer> iterator() { }

What class do we need to create to hold the iterators state?

Iterator

Where should that class be created?

private inner class → inside our collection or data structure

What interface does it need to implement?

Iterator<E> → Iterator<Integer>

What method(s) do we need to implement for that interface?

E Next();

boolean hasNext();

→ Integer next() { }

What is the process to iterate over an object?

1. save the current value into a temp variable
2. move to the next item (update state)
3. return the temp value

How could we make our MemoryStream from the next page work in an enhanced for loop? What changes would we need to make to the MemoryStream class?

```
MemoryStream<Integer> stream = new MemoryStream<Integer>();
//add data to stream
for (Integer i: stream) {
  System.out.println(i);
}
stream.close();
```

Name: ____________________ PID: __________________ Code: 1213

```
class MyClass<E> implements Iterable<E> {
    class MyIterator<E> implements Iterator<E> {
        // state
        public MyIterator(______) {
            // save initial state
        }
        public E next() {
            return null;
        }
        public boolean hasNext() {
            return false;
        }
    }
    public Iterator<E> iterator() {
        return new MyIterator(___);
    }
}
```

}

}

2nd page → should be this (see updated PDF on website)

How could we make our linked list work in an enhanced for loop? What changes would we need to make to the LList class?

```
LList<Integer> list = new LList<Integer>();

//code to add data to list

for (Integer i: list) {
  System.out.println(i);
}
```

```
Integer i;                      should be Iterator
while ( listItr.hasNext() ) {
    i = listItr.next();
    S.o.println(i);
}
```

```
public class LList<E> implements Iterable<E> {
  Node<E> front;
  int size;

  LList() { //... }
  public void prepend(E value) { //... }
  public E get(int index) { //... }
  public int size() { //... }
  public Iterator<E> iterator() {
      return new LLIterator<E>();
  }
  class LLIterator<E> implements Iterator<E> {
     //state
     Node<E> current;

     public LLIterator() {
         current = front.next;

     }
     public boolean hasNext() {
         return current != null;
     }
     public E next() {
        // throw exception if data changed
        ① E temp = current.value;
        ② current = current.next;
        ③ return temp;
     }
  }

}
```

```
class Node<E> {
  E value;
  Node<E> next;
  public Node(E value, Node<E> next) {
    this.value = value;
    this.next = next;
  }
}
```