4 Object-Oriented Programming (acr31)

(a) Describe the differences between primitive types and objects in Java. Consider:

(i) the values they contain
(ii) where they are stored in memory
(iii) how they interact with Java references

(b) What are auto-boxing and auto-unboxing? Give an example of how they might cause an exception to be thrown.

(c) Consider the following code in which any arbitrary Java type (primitive or object) could be substituted for T.

```java
void f(T t) { /* ... */ }
T t1 = /* ... */
f(t1);
```

For which substitutions of T can we guarantee that the value in t1 is unchanged after the invocation of f(t1)? Justify your answer.

(d) Explain how Java’s implementation of generics precludes substituting T with a primitive type.

(e) You are asked to redesign the standard library to incorporate an immutable list. Explain the relative merits of:

(i) MutableList being a subtype of ImmutableList
(ii) ImmutableList being a subtype of MutableList
(iii) ImmutableList and MutableList having no common supertype
(iv) ImmutableList and MutableList both subtyping CommonList