Explain what makes Scala purely object-oriented.

- Every value is an object.
- Every operation is a method call.

What problems are associated with statically typed languages?

- They are verbose.
- They prevent programmers from expressing themselves as they wish.
- They prohibit certain patterns of dynamic modification of software systems.

What are the two guiding ideas of functional programming?

- Functions are first-class values.
- Operations should map input values to output values rather than change data in place.

What is the key insight of object-orientated languages?

They make data/operation containers fully general so that they are values that can be put in other containers, or passed as parameters to operators.

How do you run Scala scripts on Unix?

```
#! /bin/sh
exec scala "$0" "$@"
!#
```

Name two Java deviations from pure object-orientation.

- Some values are not objects.
- Static fields/methods are not members of any objects.

Why are Scala type annotations after the variable?

So it can be left off for type inference. If it were before, nothing would mark the definition start and a new keyword would be needed, like C# 3's `var`.

Name three advantages of static typing.

- Verifiable properties.
- Safe refactoring.
- Documentation: static types are never out of date, IDE context help.