Language semantics

In general, the semantics of the Scala Native language are the same as Scala on the JVM. However, a few differences exist, which we mention here.

Interop extensions

Annotations and types defined `scala.scalanative.native` may modify semantics of the language for sake of interoperability with C libraries, read more about those in Native code interoperability section.

Multithreading

Scala Native doesn't yet provide libraries for parallel multi-threaded programming and assumes single-threaded execution by default.

It's possible to use C libraries to get access to multi-threading and synchronization primitives but this is not officially supported at the moment.

Undefined behavior

A number of error conditions which are well-defined on JVM are undefined behavior:

1. Dereferencing null.
2. Division by zero.
3. Stack overflows.

Those typically crash application with a segfault on the supported architectures.

Continue to Native code interoperability.