R11.5

A try-catch block will handle the exception within the method, while a throw clause will throw the exception to the caller to handle.

R11.6

A checked exception is a type of exception that must be either caught or declared in the method in which it is thrown. It is checked at compile time, e.g., java.io.IOException.

An unchecked exception is a kind of exception that can be thrown without being caught or declare. It is not checked at compile time, e.g., runtime exceptions like NullPointerException.

A checked exception needs to be declared.

R11.14

InputMismatchException will be thrown if the next token does not match the expected type or is out of range (more than the maximum value or less than the minimum value).

NoSuchElementException will be thrown if input is exhausted but nextInt() method is still invoked.

IllegalStateException will be thrown if scanner is closed and nextInt() method is still invoked.

They are all checked exceptions.

R11.17

The out.close() will always be run and if there should be an exception thrown for the constructor, the exception is not caught or handled.

Even if the constructor is moved to the try block, there are chances that an exception is caught and thrown but the out.close() still got invoked.