In computer and Java, numbers are stored in 2’s complement form:

A positive number’s 2’s complement is itself

A negative number’s 2’s complement is:

(1) Including the sign bit, reverse every bit of the positive number

(2) +1

Pos + Pos, Pos + Neg, Neg + Pos, Neg + Neg should add all bits (including the sign bit). The result is also in 2’s complement form. If sign changes, overflow happens.

What is the real value of a negative number which is currently in 2’s complement form?

1. Keep the sign bit, reverse other bit of this number
2. +1