This C++ program demonstrates bitwise operators using two numbers, `a = 2` and `b = 4`.

- **AND (`&`): Compares each bit of `a` and `b`, resulting in `0`, since no bits are `1` in the same position.
- **OR (`|`): Combines bits from `a` and `b`, resulting in `6`, which represents both numbers combined.
- **XOR (`^`): Highlights bits that are different between `a` and `b`, also resulting in `6`.
- **Left Shift (`<<`): Shifts the bits of `a` left, doubling its value to `4`.
- **Right Shift (`>>`): Shifts bits of `a` right, reducing it to `1`.
- **One’s Complement (`~`): Inverts the bits of `a`, producing `-3` as the result.

The program is a simple example of how these operators work at the binary level, which is important in low-level operations and optimizing code.