

22 Lecture - CS201

Important Subjective

1. **What is bitwise AND operator and how does it work?**

Answer: The bitwise AND operator is represented by the symbol "&". It operates on two binary numbers by performing an AND operation on each corresponding pair of bits. The result is a binary number with a 1 in each bit position where both corresponding bits were 1 in the input numbers. For example, $1101 \& 1011 = 1001$.

2. **What is the purpose of bitwise XOR operator?**

Answer: The bitwise XOR operator is represented by the symbol "^". It operates on two binary numbers by performing an XOR operation on each corresponding pair of bits. The result is a binary number with a 1 in each bit position where the corresponding bits are different in the input numbers. The purpose of this operator is to flip the bits in the output where the input bits differ, which can be useful for various purposes such as encryption.

3. **What is left shift operator and how does it work?**

Answer: The left shift operator is represented by the symbol "<<". It operates on a binary number by shifting all of its bits to the left by a specified number of positions. The result is a binary number with 0s shifted in on the right side. For example, $1010 \ll 2 = 101000$.

4. **What is the difference between signed and unsigned right shift operator?**

Answer: The signed right shift operator ">>" preserves the sign of the input number when shifting its bits to the right. The unsigned right shift operator ">>>" fills in 0s on the left side when shifting its bits to the right, regardless of the sign of the input number.

5. **What is bitwise NOT operator and how does it work?**

Answer: The bitwise NOT operator is represented by the symbol "~". It operates on a binary number by flipping all of its bits from 0 to 1 and 1 to 0. The result is the one's complement of the input number. For example, $\sim 1010 = 0101$.

6. **What is the purpose of bitwise OR operator?**

Answer: The bitwise OR operator is represented by the symbol "|". It operates on two binary numbers by performing an OR operation on each corresponding pair of bits. The result is a binary number with a 1 in each bit position where at least one corresponding bit is 1 in the input numbers. The purpose of this operator is to combine sets of binary flags or to set particular bits in a binary number.

7. **How do you perform a bitwise AND assignment operation in C++?**

Answer: To perform a bitwise AND assignment operation in C++, the "&=" operator is used. For example, if a is a variable containing the binary number 1010 and b is a variable containing the binary number 1101, the statement "a &= b;" will perform a bitwise AND operation on a and b, and store the result in a.

8. **What is the purpose of bitwise assignment operators?**

Answer: The purpose of bitwise assignment operators is to combine the operations of bitwise operators and assignment operators. These operators are used to modify the value of a variable

in place using a bitwise operation. They are a convenient and efficient way to modify individual bits of a variable.

9. **How do you perform a left shift assignment operation in Java?**

Answer: To perform a left shift assignment operation in Java, the "<<=" operator is used. For example, if a is a variable containing the binary number 1010, the statement "a <<= 2;" will shift all of the bits in a to the left by 2 positions, resulting in the binary number 101000.

10. **What is the difference between prefix and postfix increment operators in C++?**

Answer: The prefix increment operator "++