Mô phỏng một số phép toán trên hệ nhị phân: cộng, trừ, nhân, chia.

Để mô phỏng các phép toán cộng, trừ, nhân, chia trên hệ nhị phân trong C++, ta có thể sử dụng các phép toán bit (bitwise operators) và các phép toán logic (logical operators) để thực hiện các phép toán này. Dưới đây là ví dụ về cách thực hiện các phép toán này trong C++:

1. **Phép toán cộng (addition):**

0 + 0 = 0

0 + 1 = 1

1 + 0 = 1

1 + 1 = 0 (nhớ 1)

Vd : công 2 số 10001112​ và số 111102111102​

1 0 0 0 1 1 1 (71)

+ 1 1 1 1 0 (30)

---------------

= 1 1 0 0 1 0 1 (101)

**2. Phép toán trừ (subtraction):**

0 − 0 = 0

0 − 1 = 1 (mượn 1)

1 − 0 = 1

1 − 1 = 0

Vd :

1 1 0 1 1 1 0

− 1 0 1 1 1

---------------

= 1 0 1 0 1 1 1 ← kết quả

------------------------------ vd 2 :

1 1 1 1 (bit mượn)

1 1 0 1 1 1 0

- 1 0 1 1 1

---------------

= 1 0 1 0 1 1 1 ← kết quả

3. Phép toán nhân (multiplication):

1 0 1 1 (A)

× 1 0 1 0 (B)

----------------

0 0 0 0 ← tương đương với 0 trong B

+ 1 0 1 1 ← tương đương với 1 trong B

+ 0 0 0 0

+ 1 0 1 1

----------------

= 1 1 0 1 1 1 0

4. Phép toán chia (division):

Vd 1 :

159 | 12

- 12 |----

-- | 13

39 |

-36 |

-- |

3 |

Vd 2:

10011111 | 1100

- 1100 |------

---- | 1101

1111 |

- 1100 |

---- |

1111 |

- 1100 |

---- |

11 |

**5. Viết đoạn chương trình C++ để thử nghiệm và trả lời các kết quả sau:**

**a. Khi gán x=a, với a là hằng số vượt ra ngoại phạm vi có thể chứ của biến nguyên x**

**thì kết quả nhận được sẽ như thế nào?**

**#include <iostream>**

**int main() {**

**int x = 5;**

**std::cout << "x = " << x << std::endl;**

**x = 10000000000; // giá trị vượt quá phạm vi của kiểu int**

**std::cout << "x = " << x << std::endl;**

**return 0;**

**}**