

Please select the practice item: 13

User: Adrian Lozada

In this exercise, you need to compute  $66 + (-30)$  using 8-bit 1's complement operation. Verify if the result is correct.

Step 1: Convert decimal data to binary

Please convert 66 to signed decimal to 8-bit binary representation with 1's complement: 01000010

Please convert -30 to signed decimal to 8-bit binary representation with 1's complement: 11100001

Step 2a: Perform binary addition

```
  0 1 0 0 0 0 1 0
+ 1 1 1 0 0 0 0 1
-----
```

Please input the final carry bit (0 or 1)?: 1

Please input the above binary addition result: 00100011

```
  0 1 0 0 0 0 1 0
+ 1 1 1 0 0 0 0 1
-----
 1 0 0 1 0 0 0 1 1
```

Step 2b: End-Round Carry?

End-round carry operation is required for this binary addition? (y/n): y

```
  0 0 1 0 0 0 1 1
+ 0 0 0 0 0 0 0 1
-----
```

Please input the final carry bit (0 or 1)?: 00100100

Please input the above binary addition result: 00100100

```
  0 0 1 0 0 1 0 0
+ 0 0 0 0 0 0 0 1
-----
 0 0 1 0 0 1 0 0
```

Step 3: Verify the result.

Please convert the result binary sum to decimal value: 36

Overflow occurs or not (y/n)?: n

The final result is correct and no overflow occurs.

Well done!

Please select the practice item: 13

User: Adrian Lozada

In this exercise, you need to compute  $99 + (-15)$  using 8-bit 1's complement operation. Verify if the result is correct.

Step 1: Convert decimal data to binary

Please convert 99 to signed decimal to 8-bit binary representation with 1's complement: 01100011

Please convert -15 to signed decimal to 8-bit binary representation with 1's complement: 11110000

Step 2a: Perform binary addition

```
  0 1 1 0 0 0 1 1
+ 1 1 1 1 0 0 0 0
-----
```

Please input the final carry bit (0 or 1)?: 1

Please input the above binary addition result: 01010011

```
  0 1 1 0 0 0 1 1
+ 1 1 1 1 0 0 0 0
-----
 1 0 1 0 1 0 0 1 1
```

Step 2b: End-Round Carry?

End-round carry operation is required for this binary addition? (y/n): y

```
  0 1 0 1 0 0 1 1
+ 0 0 0 0 0 0 0 1
-----
```

Please input the final carry bit (0 or 1)?: 01010100

Please input the above binary addition result: 01010100

```
  0 1 0 1 0 1 0 0
+ 0 0 0 0 0 0 0 1
-----
 0 1 0 1 0 1 0 0
```

Step 3: Verify the result.

Please convert the result binary sum to decimal value: 84

Overflow occurs or not (y/n)?: n

The final result is correct and no overflow occurs.

Well done!

Please select the practice item: 14

User: Adrian Lozada

In this exercise, you need to compute  $21 + 5$  using 8 - bit 2's complement operation. Verify if the result is correct.

Step 1: Convert decimal data to binary

Please convert 21 to signed decimal to 8-bit binary representation with 2's complement: 00010101

Please convert 5 to signed decimal to 8-bit binary representation with 2's complement: 00000101

Step 2: Perform binary addition

```
  0 0 0 1 0 1 0 1
+ 0 0 0 0 0 1 0 1
-----
```

Please input the final carry bit (0 or 1)?: 0

Please input the above binary addition result: 00011010

```
  0 0 0 1 0 1 0 1
+ 0 0 0 0 0 1 0 1
-----
  0 0 0 1 1 0 1 0
```

Step 3: Verify the result.

Please convert the result binary sum to decimal value: 26

Overflow occurs or not (y/n)?: n

The final result is correct and no overflow occurs.

Well done!

Please select the practice item: 14

User: Adrian Lozada

In this exercise, you need to compute  $44 + (-10)$  using 8 - bit 2's complement operation. Verify if the result is correct.

Step 1: Convert decimal data to binary

Please convert 44 to signed decimal to 8-bit binary representation with 2's complement: 00101100

Please convert -10 to signed decimal to 8-bit binary representation with 2's complement: 11110110

Step 2: Perform binary addition

```
  0 0 1 0 1 1 0 0
+ 1 1 1 1 0 1 1 0
-----
```

Please input the final carry bit (0 or 1)?: 1

Please input the above binary addition result: 00100010

```
  0 0 1 0 1 1 0 0
+ 1 1 1 1 0 1 1 0
-----
  1 0 0 1 0 0 0 1 0
```

Step 3: Verify the result.

Please convert the result binary sum to decimal value: 34

Overflow occurs or not (y/n)?: n

The final result is correct and no overflow occurs.

Well done!