

EXPERIMENT 11: BITWISE OPERATORS

Experiment 1 Bitwise AND, OR, NOT Operators

Aim

To perform bitwise AND, OR, and NOT operations on integer values.

Algorithm

1. Read two integers
2. Apply bitwise AND
3. Apply bitwise OR
4. Apply bitwise NOT
5. Display results

Pseudocode

```
READ a, b
AND = a & b
OR = a | b
NOT a = ~a
DISPLAY results
```

Flowchart (ASCII)

```
Start
  ↓
Read a, b
  ↓
AND / OR / NOT
  ↓
Display Result
  ↓
End
```

C Program

```
#include <stdio.h>

int main() {
    int a, b;

    printf("Enter two integers: ");
    scanf("%d %d", &a, &b);

    printf("Bitwise AND (a & b): %d\n", a & b);
    printf("Bitwise OR  (a | b): %d\n", a | b);
    printf("Bitwise NOT (~a): %d\n", ~a);

    return 0;
}
```

Output

```
PS C:\Users\ASUS\Desktop\Dahadi\class c> gcc t.c
PS C:\Users\ASUS\Desktop\Dahadi\class c> ./a.exe
Enter two integers: 3
2
Bitwise AND (a & b): 2
Bitwise OR (a | b): 3
Bitwise NOT (~a): -4
PS C:\Users\ASUS\Desktop\Dahadi\class c> 
```

Conclusion

Bitwise operators perform operations at the bit level and are useful for low-level programming.

Experiment 2: Left Shift and Right Shift Operators

Aim

To demonstrate left shift and right shift bitwise operators in C.

Algorithm

1. Read an integer
2. Apply left shift operator
3. Apply right shift operator
4. Display the results

Pseudocode

```
READ num
LEFT = num << 1
RIGHT = num >> 1
DISPLAY results
```

Flowchart (ASCII)

```
Start
  ↓
Read Number
  ↓
Left Shift / Right Shift
  ↓
Display Result
  ↓
End
```

C Program

```
#include <stdio.h>

int main() {
```

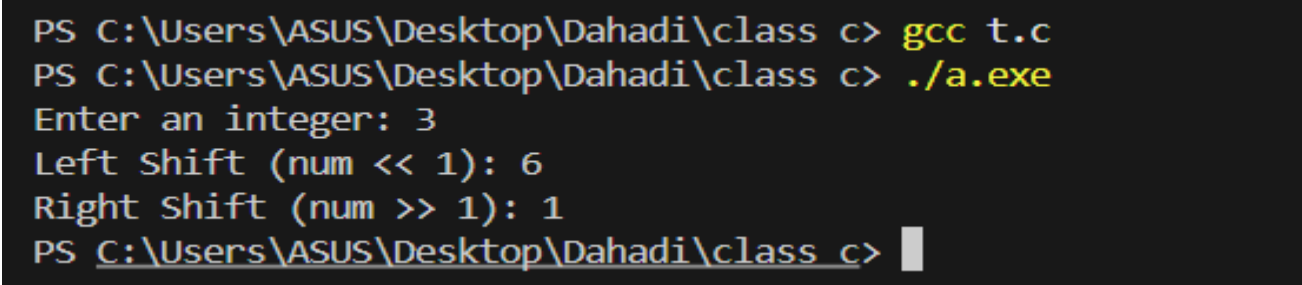
```
int num;

printf("Enter an integer: ");
scanf("%d", &num);

printf("Left Shift (num << 1): %d\n", num << 1);
printf("Right Shift (num >> 1): %d\n", num >> 1);

return 0;
}
```

Output



```
PS C:\Users\ASUS\Desktop\Dahadi\class c> gcc t.c
PS C:\Users\ASUS\Desktop\Dahadi\class c> ./a.exe
Enter an integer: 3
Left Shift (num << 1): 6
Right Shift (num >> 1): 1
PS C:\Users\ASUS\Desktop\Dahadi\class c> █
```

Conclusion

Shift operators move bits left or right, effectively multiplying or dividing the number by powers of 2.