Efficient way to multiply with 7

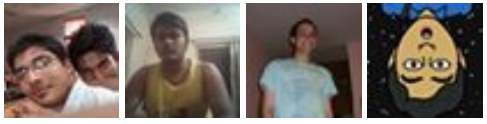
We can multiply a number by 7 using bitwise operator. First left shift the number by 3 bits (you will get 8n) then subtract the original numberfrom the shifted number and return the difference (8n – n).

Program:

```
# include<stdio.h>

int multiplyBySeven(unsigned int n)
{
    /* Note the inner bracket here. This is needed
       because precedence of '-' operator is higher
       than '<<' */
    return ((n<<3) - n);
}

/* Driver program to test above function */
int main()
{
    unsigned int n = 4;
    printf("%u", multiplyBySeven(n));

    getchar();
    return 0;
}
```

Time Complexity: O(1)
Space Complexity: O(1)

Note: Works only for positive integers.



Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

Same concept can be used for fast multiplication by 9 or other numbers.

Related Tpoics:

- Check if a number is multiple of 9 using bitwise operators
- How to swap two numbers without using a temporary variable?
- Divide and Conquer | Set 4 (Karatsuba algorithm for fast multiplication)
- Find position of the only set bit
- Swap all odd and even bits
- Add two bit strings
- Write your own strcmp that ignores cases
- Binary representation of a given number

Writing code in comment? Please use **ideone.com** and share the link here.

7 Comments GeeksforGeeks

Sort by Newest

Popular Posts

- All permutations of a given string
- Memory Layout of C Programs
- Understanding “extern” keyword in C
- Median of two sorted arrays
- Tree traversal without recursion and without stack!
- Structure Member Alignment, Padding and Data Packing
- Intersection point of two Linked Lists
- Lowest Common Ancestor in a BST.
- Check if a binary tree is BST or not
- Sorted Linked List to Balanced BST

Join the discussion…

Nimish • 7 months ago

why not n+(n<<1)+(n<<2) ?
as bit shifts are still three only, though not don at once, but instead of one "-" w
using "+" can be more efficient than using "-" i guess.

1 ^ | v •

vindhya • 2 years ago

someone plz explain this..how does it give O(1) complexity??sorry i m weak a

```
/* Paste your code here (You may delete these lines if not writing co
```

^ | v •

vindhya ➚ vindhya • 2 years ago

oh..!!understood..:)

```
/* Paste your code here (You may delete these lines if not writ
```

^ | v •

frankc • 3 years ago

Is it possible to use divide and conquer and the "clever" multiplication algorithm
by 7 or 15?

/* Paste your code here (You may delete these lines if not writing co

705

Subscribe

bala • 3 years ago

Sometimes ,we do have to take care of overflow even if multiplication by 7 or 1

nitesh.gami008 bala • 3 years ago

@bala....Please give some example....

adsf nitesh.gami008 • a year ago

Consider this binary number
0100 0000
how will you shift 3 times without handling overflow

Correct me if i am wrong

Subscribe Add Disqus to your site

Recent Comments

Abhi You live US or India?
Google (Mountain View) interview · 17 minutes ago

Aman Hi, Why arent we checking for conditions...
Write a C program to Delete a Tree. · 57 minutes ago

kzs please provide solution for the problem...
Backtracking | Set 2 (Rat in a Maze) · 1 hour ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...
Root to leaf path sum equal to a given number · 1 hour ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...
Count trailing zeroes in factorial of a number · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...
Find subarray with given sum · 1 hour ago