GeeksforGeeks
A computer science portal for geeks

GeeksQuiz

Login

Home   Algorithms   DS   GATE   Interview Corner   Q&A   C   C++   Java   Books   Contribute   Ask a Q   About

Array   Bit Magic   C/C++   Articles   GFacts   Linked List   MCQ   Misc   Output   String   Tree   Graph

Binary representation of a given number

Write a program to print Binary representation of a given number.

Source: Microsoft Interview Set-3

Method 1: Iterative
For any number, we can check whether its 'i'th bit is 0(OFF) or 1(ON) by bitwise ANDing it with "2^i" (2 raise to i).

```
1) Let us take number 'NUM' and we want to check whether it's 0th bit is ON or OFF

      bit = 2 ^ 0 (0th bit)
      if  NUM & bit == 1 means 0th bit is ON else 0th bit is OFF

2) Similarly if we want to check whether 5th bit is ON or OFF

      bit = 2 ^ 5 (5th bit)
      if NUM & bit == 1 means its 5th bit is ON else 5th bit is OFF.
```

Let us take unsigned integer (32 bit), which consist of 0-31 bits. To print binary representation of unsigned integer, start from 31th bit, check whether 31th bit is ON or OFF, if it is ON print "1″ else print "0″. Now check whether 30th bit is ON or OFF, if it is ON print "1″ else print "0″, do this for all bits from 31 to 0, finally we will get binary representation of number.

```c
void bin(unsigned n)
{
    unsigned i;
    for (i = 1 << 31; i > 0; i = i / 2)
        (n & i)? printf("1"): printf("0");
}

int main(void)
```

Interview Experiences

Advanced Data Structures

Dynamic Programming

Greedy Algorithms

Backtracking

Pattern Searching

Divide & Conquer

Mathematical Algorithms

Recursion

Geometric Algorithms

```
{
    bin(7);
    printf("\n");
    bin(4);
}
```

Method 2: Recursive

Following is recursive method to print binary representation of 'NUM'.

```
step 1) if NUM > 1
        a) push NUM on stack
        b) recursively call function with 'NUM / 2'
step 2)
        a) pop NUM from stack, divide it by 2 and print it's remainder.
```

```c
void bin(unsigned n)
{
    /* step 1 */
    if (n > 1)
        bin(n/2);

    /* step 2 */
    printf("%d", n % 2);
}

int main(void)
{
    bin(7);
    printf("\n");
    bin(4);
}
```

This article is compiled by **Narendra Kangralkar**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Popular Posts

All permutations of a given string

Memory Layout of C Programs
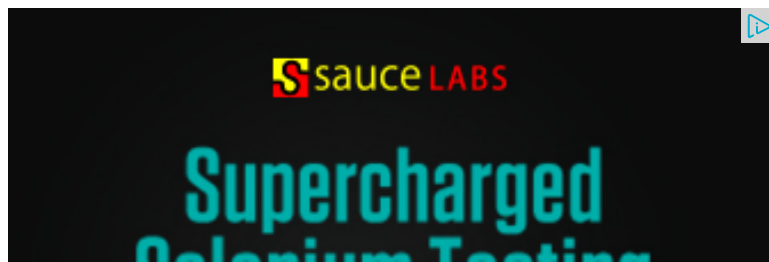
Understanding "extern" keyword in C

Median of two sorted arrays

Tree traversal without recursion and without stack!

Structure Member Alignment, Padding and Data Packing
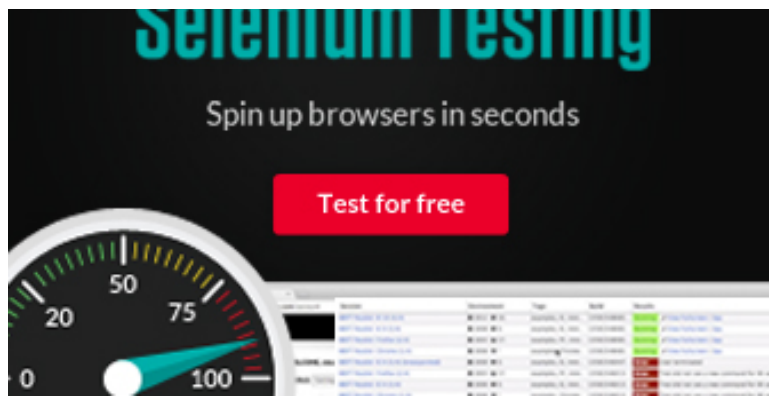
Intersection point of two Linked Lists

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
- Find the element that appears once

| 6 | **Tweet** 1 | 0 |

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

22 Comments   **GeeksforGeeks**

Sort by Newest ▾

Join the discussion…

Raj · 5 months ago

Hai frds I have got a new logic .. see it http://www.codextream.com/?p=5...

this works for both 16-bit and 32-bit word

∧ | ∨ · Reply · Share ›

groomnestle · 5 months ago

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

char *binary(int n)

{

char *bin = (char*)malloc(33);

memset(bin,0,33);

int i=31;

int j;

for(j=0;j<32;j++)

{

bin[j]=((1<<i)&n)?'1':'0'; i--;="" }="" return="" bin;="" }="" int="" main()="" {="" int
binary(10));="" return="" 0;="" }="">

∧ | ∨ · Reply · Share ›

neelabhsingh · 6 months ago

Method 2: Recursive this method can not be use for -ve number

atiq · 10 months ago

// For negative/positive .... 4byte representation

```
 /*
#include<iostream>
using namespace std;
void Binary(signed int n)
{  char result[33];
        int i=31;
         result[32]='&#92&#48';
        signed int k=1;
        while(i+1)
        {
                if(k&n)
                {
                        result[i]=49;
                }
                else
                {
```

see more

∧ | ∨ · Reply · Share ›

neelabhsingh → atiq · 6 months ago

Above code will work for all types number -ve or positive........
because i is taken as unsigned and it will always positive and u can us
and +ve number.......

#include<stdio.h>
void showbit(int num,int size)

```
unsigned int i=1<<size-1; while(i="">=1)
{
(i&num)? printf("1"):printf("0");
i=i/2;
}
}
int main()
{
int n;
printf("Enter any digits\n");
scanf("%d",&n);
showbit(n,sizeof(int)*8);
return 0;
}
```

Lomesh Meshram · 10 months ago

```
#include<stdio.h>
int main()
{
int num;
unsigned int i=0x80000000;.
int size=sizeof(num)*8;
scanf("%d",&num);
printf("n Binary Representation is------n");.
while(size--)
{
if(num & i).

printf("1");.
else
```

```
    printf("0");.
    i=i>>1;.
    }
    printf("n");
    return 0;
    }
```

∧ | ∨ • Reply • Share ›

Dj • a year ago

Above code works fine .... but
if((x&(0x80000000))>0) condition should work for <0
because
if first bit of x is 1, x&0x80000000 gives negative number -2147483648,which i:

∧ | ∨ • Reply • Share ›

Dj • a year ago

```
 #include<stdio.h>

int main()
{
    int x;
            scanf("%d",&x);
            x=0x80000000;
            printf("%d\n",x);
            for(int i=0;i<32;i++)
            {
                    if((x&(0x80000000))>0)
                                printf("1");
                    else
                        printf("0");
                    x=x<<1;
            }
```

Are you a developer? Try out the HTML to PDF API

```
        return 0;
    }
```

Above code works fine .... but
if((x&(0x80000000))>0) condition should work for <0
because
if first bit of x is 1, x&0x80000000 gives negative number -2147483648,which i

abhishek08aug · a year ago

My code:

```
 #include<stdio.h>
 #include<stdlib.h>

char * bit_representation(unsigned int num) {
  char * bit_string = (char *)malloc(sizeof(char)*sizeof(unsigned int
  unsigned int i=1, j;
  for(i=i<<(sizeof(unsigned int)*8-1), j=0; i>0; i=i>>1, j++) {
    if(num&i) {
      *(bit_string+j)='1';
    } else {
      *(bit_string+j)='0';
    }
  }
  *(bit_string+j)='&#092&#048';
  return bit_string;
}
```

see more

Amandeep Sharma · a year ago

This method works for both +ve and -ve numbers

```c
 #include<stdio.h>
 #include<conio.h>
void showbits(int n)
{
        int mask=0x80000000;
        int format=0;//just for formating
        if(mask & n) printf("1");
        else printf("0");
        mask=0x40000000;//remember i have not use here mask=mask>>1 b
        while(mask)
        {
                if(mask & n) printf("1");
                else printf("0");
                if(s=(++s%2))printf(" ");//for formatting
                mask=mask>>1;
        }
}
```

see more

⌃ | ⌄ · Reply · Share ›

Seelam Komalkumar · a year ago

will it takes care of negitive numbers?

⌃ | ⌄ · Reply · Share ›

ASHISH · a year ago

```c
void dec_to_bin(int n)
{
```

```
        int rem,idx =7;
        int count[8]={0};
        while(n>1)
        {
            rem = n%2;
            cout<<"rem="<<rem<<endl;
            n = n/2;
            count[idx--]=rem;
        }
        count[idx]=n;
        for(int i=0;i<8;i++)
            cout<<count[i];
    }
```

∧ | ∨ • Reply • Share ›

Frank · a year ago

Please, guys, when you only want to print a char, no need to use printf !
putchar() is largely sufficient !

∧ | ∨ • Reply • Share ›

Kanhaiya · a year ago

I think it will print the binary digits in reverse order. we need to use stack to prir

∧ | ∨ • Reply • Share ›

Venki ➜ Kanhaiya · a year ago

@Kanhaiya, it will not print in reverse order. It prints from MSB to LSB.
the same.

∧ | ∨ • Reply • Share ›

Agreed Venki. I was just thinking about that usually we write LSI

^ | ⌄ · Reply · Share ›

Pintu Gupta · a year ago
#include
using namespace std;
void binary_represent(unsigned);
int main()
{
unsigned num;
cin>>num;

binary_represent(num);
return 0;
}

void binary_represent(unsigned num)
{
if(!num)
{
return;
}
binary_represent(num>>1);
cout<<(num&1);

}

^ | ⌄ · Reply · Share ›

neelabhsingh → Pintu Gupta · 6 months ago
int main()
{
int num;

Are you a developer? Try out the HTML to PDF API

```cpp
cin>> num;
int i=1>>31;
while(i>=1)
{
(i&num)? cout<< "1" : cout<< "0";
i=i/2;

}

}
```

▲ | ▼   ·   Reply   ·   Share ›

mahendra singh ➔ Pintu Gupta · a year ago

```cpp
#include
#include

void ConvertBinaryStr(unsigned int uiNum, char pcBinStr);

int main()
{
unsigned int uiNum;
char acBinRep[33]="";

clrscr();

printf("Enter Number\n");
scanf("%u", &uiNum);

ConvertBinaryStr(uiNum, acBinRep);
printf("Binary Representation: %s\n", acBinRep);
getch();

return 0;
```

see more

open in browser PRO version   Are you a developer? Try out the HTML to PDF API   pdfcrowd.com

∧ | ∨ · Reply · Share ›

OP · a year ago

iterative program is cool :)

2 ∧ | ∨ · Reply · Share ›

neelabhsingh → OP · 6 months ago

No it will not work for negative number.......

∧ | ∨ · Reply · Share ›

Anonymous bin Ich → neelabhsingh · a month ago

That is not true. Just change
void bin(unsigned int n) to
void bin(int n)

The problem is mitigated because we use unsigned i.

∧ | ∨ · Reply · Share ›

✉ Subscribe   Ⓓ Add Disqus to your site