

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  /*
4   Exponent (m)^n
5   power:
6
7   2^5 = 2 * 2 * 2 * 2 * 2
8
9   (m)^n = m * m * m * ..... * (n-1) times * m
10
11  pow(m, n) = pow(m, n-1) * m;
12  */
13  int pow(int m, int n)
14  {
15      if(n == 0)
16          return 1;
17      else{
18
19          return pow(m, n-1) * m;
20      }
21  }
22  }
23  /*
24      pow(2, 5)    => 32
25      |
26      pow(2, 4) * 2 => 16 * 2
27      |
28      pow(2, 3) * 2 => 8 * 2
29      |
30      pow(2, 2) * 2 => 4 * 2
31      |
32      pow(2, 1) * 2 => 2 * 2
33      |
34      pow(2, 0) * 2 => 1 * 2
35      |
36      x return 1
37
38  */
39  int powM(int m, int n)
40  {
41      if(n == 0)
42          return 1;
43      if(n % 2 == 0)
44      {
45          return powM(m*m, n/2);
46      }
47      else{
48          return powM(m*m, (n-1)/2) * m;
49      }
50  }
51  }
52  /*
53      pow(2, 5)    2^5 => 32

```

```

54     |
55     pow(2*2, 4/2) * 2;    => 2^4 * 2
56     |
57     pow(2^2 * 2^2, 2/2 ); => 2^4
58     |
59     pow(2^4 * 2^4, 0/2) * 2^4; => 1 * 2^4
60     |
61     return 1
62
63 */
64 int main()
65 {
66     int m, n;
67     m = 2;
68     n = 5;
69
70     cout << pow(m, n);
71
72     cout << powM(m, n);
73
74 }

```