|  |  |
| --- | --- |
| **Power of 2 in C++** | |
| #include <iostream>  using namespace std;  void powerOf2(int x) {  if ((x & (x - 1)) == 0) {  cout << x << " is Power of two" << endl;  } else {  cout << x << " is not Power of two" << endl;  }  }  int main() {  int x = 9;  for (int i = 1; i <= 32; i++) {  powerOf2(i);  }  return 0;  } | **Key Logic:**  if ((x & (x - 1)) == 0)  This works because:   * A power of two has only **one set bit** in binary. * x & (x - 1) turns off the lowest set bit, so:   + If result is 0 → x was a power of 2.   + Otherwise → it's not.   **🔢 Dry Run Table (for x from 1 to 16 for brevity):**   | **x** | **Binary of x** | **x-1** | **Binary of x-1** | **x & (x-1)** | **Is Power of 2?** | | --- | --- | --- | --- | --- | --- | | 1 | 00000001 | 0 | 00000000 | 00000000 | ✅ Yes | | 2 | 00000010 | 1 | 00000001 | 00000000 | ✅ Yes | | 3 | 00000011 | 2 | 00000010 | 00000010 | ❌ No | | 4 | 00000100 | 3 | 00000011 | 00000000 | ✅ Yes | | 5 | 00000101 | 4 | 00000100 | 00000100 | ❌ No | | 6 | 00000110 | 5 | 00000101 | 00000100 | ❌ No | | 7 | 00000111 | 6 | 00000110 | 00000110 | ❌ No | | 8 | 00001000 | 7 | 00000111 | 00000000 | ✅ Yes | | 9 | 00001001 | 8 | 00001000 | 00001000 | ❌ No | | 10 | 00001010 | 9 | 00001001 | 00001000 | ❌ No | | 11 | 00001011 | 10 | 00001010 | 00001010 | ❌ No | | 12 | 00001100 | 11 | 00001011 | 00001000 | ❌ No | | 13 | 00001101 | 12 | 00001100 | 00001100 | ❌ No | | 14 | 00001110 | 13 | 00001101 | 00001100 | ❌ No | | 15 | 00001111 | 14 | 00001110 | 00001110 | ❌ No | | 16 | 00010000 | 15 | 00001111 | 00000000 | ✅ Yes |   **✅ Output for i = 1 to 32:**  The function will print:  1 is Power of two  2 is Power of two  3 is not Power of two  4 is Power of two  5 is not Power of two  ...  32 is Power of two |
| Output:-  1 is Power of two  2 is Power of two  3 is not Power of two  4 is Power of two  5 is not Power of two  6 is not Power of two  7 is not Power of two  8 is Power of two  9 is not Power of two  10 is not Power of two  11 is not Power of two  12 is not Power of two  13 is not Power of two  14 is not Power of two  15 is not Power of two  16 is Power of two  17 is not Power of two  18 is not Power of two  19 is not Power of two  20 is not Power of two  21 is not Power of two  22 is not Power of two  23 is not Power of two  24 is not Power of two  25 is not Power of two  26 is not Power of two  27 is not Power of two  28 is not Power of two  29 is not Power of two  30 is not Power of two  31 is not Power of two  32 is Power of two | |