|  |  |
| --- | --- |
| **Subsets in C++** | |
| #include <iostream>  using namespace std;  int main() {  int n = 4;  for (int b = 0; b < (1 << n); b++) {  cout << b << endl;  }  return 0;  } | You're generating all numbers from 0 to 2^n - 1 using bit manipulation!  **🔍 Breakdown:**   * n = 4 → total combinations = 2^4 = 16 * (1 << n) means 1 shifted left n times → equals 2^n * Loop runs from 0 to 15, printing each value   **🧮 Dry Run Table:**   | **b** | **Binary of b** | | --- | --- | | 0 | 0000 | | 1 | 0001 | | 2 | 0010 | | 3 | 0011 | | 4 | 0100 | | 5 | 0101 | | 6 | 0110 | | 7 | 0111 | | 8 | 1000 | | 9 | 1001 | | 10 | 1010 | | 11 | 1011 | | 12 | 1100 | | 13 | 1101 | | 14 | 1110 | | 15 | 1111 | |
| Output:-  0  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | |