|  |  |
| --- | --- |
| **Generate Binary in C++** | |
| #include <iostream>  #include <queue>  #include <vector>  using namespace std;  vector<string> generate(int N) {  vector<string> ans;  queue<string> q;  q.push("1");  while (N-- > 0) {  string rem = q.front();  q.pop();  ans.push\_back(rem);  q.push(rem + "0");  q.push(rem + "1");  }  return ans;  }  int main() {  int N = 5;  vector<string> binaryNumbers = generate(N);  for (string num : binaryNumbers) {  cout << num << endl;  }  return 0;  } | **Goal:**  Generate the first N binary numbers (as strings) from 1 to the binary representation of N.  **⚙️ Algorithm Overview:**   * Use a **queue** to build binary numbers level-by-level (like a binary tree). * Start with "1", then append "0" and "1" to each popped string. * Do this N times.   **🔁 Dry Run for N = 5**   | **Iteration** | **Queue (Before Pop)** | **Popped (rem)** | **Added to Result** | **Queue (After Push)** | | --- | --- | --- | --- | --- | | 1 | ["1"] | "1" | "1" | ["10", "11"] | | 2 | ["10", "11"] | "10" | "10" | ["11", "100", "101"] | | 3 | ["11", "100", "101"] | "11" | "11" | ["100", "101", "110", "111"] | | 4 | ["100", "101", "110", "111"] | "100" | "100" | ["101", "110", "111", "1000", "1001"] | | 5 | ["101", "110", "111", "1000", "1001"] | "101" | "101" | ["110", "111", "1000", "1001", "1010", "1011"] |   **📤 Final Output:**  1  10  11  100  101 |
| 1  10  11  100  101 | |