|  |  |
| --- | --- |
| **Kth number in C++** | |
| #include <iostream>  #include <queue>  #include <string>  using namespace std;  string kth(int k) {  queue<string> q;  q.push("1");  q.push("2");  string ans;  for (int i = 0; i < k; i++) {  string temp = q.front();  q.pop();  ans = temp;  q.push(temp + "1");  q.push(temp + "2");  }  return ans;  }  int main() {  int k = 5;  cout << kth(k) << endl;  return 0;  } | ****Initial Setup:**** queue<string> q;  q.push("1");  q.push("2");  Initial queue: ["1", "2"] Dry Run Table:  | **Iteration (i)** | **Queue Before** | **temp (popped)** | **ans** | **Queue After Push** | | --- | --- | --- | --- | --- | | 0 | ["1", "2"] | "1" | "1" | ["2", "11", "12"] | | 1 | ["2", "11", "12"] | "2" | "2" | ["11", "12", "21", "22"] | | 2 | ["11", "12", "21", "22"] | "11" | "11" | ["12", "21", "22", "111", "112"] | | 3 | ["12", "21", "22", "111", "112"] | "12" | "12" | ["21", "22", "111", "112", "121", "122"] | | 4 | ["21", "22", "111", "112", "121", "122"] | "21" | **"21"** | ["22", "111", "112", "121", "122", "211", "212"] |  📤 Final Output: cout << kth(5);  Since index starts at 0, on the **5th iteration** (i = 4), we return:  21 💡 Output: 21 |
| 21 | |