|  |  |
| --- | --- |
| **Josephus in C++** | |
| #include <iostream>  using namespace std;  int p(int n) {  int i = 1;  while (i \* 2 <= n) {  i = i \* 2;  }  return i;  }  int s(int n) {  int h = p(n);  int l = n - h;  return 2 \* l + 1;  }  int main() {  int n = 5;  cout << s(n) << endl;  return 0;  } | You have two functions:   1. p(n) — Finds the **largest power of 2 less than or equal to n**. 2. s(n) — Computes 2 \* (n - p(n)) + 1.  📊 **Dry Run for** n = 5Step 1: p(5) int i = 1;  while (i \* 2 <= n) {  i = i \* 2;  }   | **i (before loop)** | **i \* 2** | **i (after loop)** | | --- | --- | --- | | 1 | 2 | 2 | | 2 | 4 | 4 | | 4 | 8 (>5) | loop exits |   ✅ p(5) = 4 Step 2: s(5) int h = p(5); // h = 4  int l = 5 - 4 = 1;  return 2 \* l + 1 = 2 \* 1 + 1 = 3;  ✅ Output: 3 |
| 3 | |