|  |  |
| --- | --- |
| **Remove adjacent duplicate in C++** | |
| #include <iostream>  #include <stack>  #include <string>  using namespace std;  string removeAdjacentDuplicates(string s) {  stack<char> st;  for (char ch : s) {  if (!st.empty() && st.top() == ch) {  st.pop();  } else {  st.push(ch);  }  }  string result = "";  while (!st.empty()) {  result = st.top() + result;  st.pop();  }  return result;  }  int main() {  string s = "abbaca";  cout << removeAdjacentDuplicates(s) << endl; // Output: "ca"  return 0;  } | Input: string s = "abbaca"; 🧠 Step-by-Step Stack Trace:  | **Step** | **Char** | **Stack (top to bottom)** | **Action** | | --- | --- | --- | --- | | 1 | 'a' | [a] | Push | | 2 | 'b' | [a, b] | Push | | 3 | 'b' | [a] | 'b' == top → Pop | | 4 | 'a' | [] | 'a' == top → Pop | | 5 | 'c' | [c] | Push | | 6 | 'a' | [c, a] | Push |  ✅ Final Stack (bottom to top): c a So result = "ca". |
| ca | |