Strings

Q-2 String is a palindrome or not.

i/p → abcba
o/p → Yes

A palindrome can be defined as if on reading from back. is giving the same string as reading from the start / front.
For example → aba is same if we read from back or front.
Here we will be using the 2 pointer approach. If while traversing, we find any equal value we will return false.
After traversing, if false was not returned then we simply can return true.

Dry run

1) "abcba" } s
i ↑ ↑ j
i = 0, j = s.length() - 1
i = 0, j = 4
i <= j ⇒ True
s[i] == s[j] also hence simply increment i & decrement j.

2) "abcba"
i ↑ ↑ j
i = 1, j = 3
i <= j ⇒ True
s[i] == s[j] also hence increment the value of i & decrement the value of j.

3) "abcba"

i → ← j

i = 2, j = 2

i <= j ⇒ True

s[i] == s[j] also hence increment i & decrement j.

Now i > j and hence return false was not encountered and hence true will be returned.

Code

```
bool isPalindrome (string &s) {
    //2 pointers are maintained
    int i=0;
    int j = s.length() - 1;
  // Run while loop
  while (i<=j) {
      if (s[i] != s[j]){
              return false; //Not Palindrome
      }
      i++; //Increment i
      j--; //Decrement j
  }
    return true; //Palindrome
}
```

Time complexity = O(n)

Space complexity = O(1)