1. 1. v1 = 7; v2 = 4; v3 =11

7

11

12

9

8

🡨 still there but will be overwritten on next push

Top = 4

1. 1. (nothing changed because 7%2 ==1

7

4

12

9

8

1. 1. v1 = 7; v2 = 4; v3 = 11

4

4

7

11

[0] [1] [2] [3] [4] [5]

Front = 5

Rear = 1

2

b.

4

4

7

4

4

7

[0] [1] [2] [3] [4] [5]

Front = 0

Rear = 3

3.

bool isPalindrome()

{

    char stackChar = '\*';

    char queueChar = '\*';

    bool b1; //Not really necessary for this program

    Stack s1;

    Queue q1;

    // read the word from cin

    do {

        cin >> stackChar; //Using stackchar because I can't create another variable. This could be any char variable

        s1.push(stackChar);

        q1.enqueue(stackChar);

    } while (stackChar != ' ');

    // pull the individual letters back out and compare

    while (!q1.isEmpty())

    {

        s1.pop(stackChar);

        q1.dequeue(queueChar);

        if (stackChar != queueChar) return false; //If even one pair does not match, the word is not a palindrome

    }

    return true;

}