



# Stacks – 2

## Assignment Solutions

## 1. Valid Parentheses [Leetcode - 20]

**Solution:**

```
class Solution {
public:
    bool isValid(string s) {
        if (s.size() % 2) return false;
        stack<char> open, close;
        for (int i = 0; i < s.size(); i++) {
            if (s[i] == '(' || s[i] == '{' || s[i] == '[')
                open.push(s[i]);
            if (!open.empty() &&
                ((s[i] == ')' && open.top() == '(') ||
                 (s[i] == '}' && open.top() == '{') ||
                 (s[i] == ']' && open.top() == '[')))
                close.push(s[i]), open.pop();
        }
        if (close.size() * 2 == s.size()) return true;
        return false;
    }
};
```

## 2. Next Greater Node in Linked List [Leetcode - 1019]

**Solution:**

```
vector<int> nextLargerNodes(ListNode* head) {
    stack<ListNode*> st;
    ListNode* temp = head;
    while (temp != NULL) {
        while (!st.empty() && (st.top())->val < temp->val) {
            ListNode* help = st.top();
            st.pop();
            help->val = temp->val;
        }
        st.push(temp);
        temp = temp->next;
    }
    while (!st.empty()) {
        ListNode* help = st.top();
        st.pop();
        help->val = 0;
    }
    vector<int> ans;
    while (head != NULL) {
        ans.push_back(head->val);
        head = head->next;
    }
    return ans;
}
```

## 3. Final Prices with a Special Discount in a Shop [Leetcode - 1475]

**Solution:**

```
class Solution {
public:
    vector<int> finalPrices(vector<int>& prices) {
        int n = prices.size();
        stack<int> st;
        vector<int> ans(n);
        for (int i = n - 1; i ≥ 0; i--) {
            while (!st.empty() and prices[st.top()] > prices[i]) {
                st.pop();
            }
            if (st.empty()) {
                ans[i] = prices[i];
            } else {
                ans[i] = prices[i] - prices[st.top()];
            }
            st.push(i);
        }
        return ans;
    }
};
```

## 4. Next Greater Element II [Leetcode - 503]

**Solution:**

```
class Solution {
public:
    vector<int> nextGreaterElements(vector<int>& nums) {
        int n = nums.size();
        vector<int> a(2 * n);
        for (int i = 0; i < 2 * n; i++) {
            a[i] = nums[i % n];
        }
        stack<int> s;
        vector<int> ans(2 * n);
        for (int i = 2 * n - 1; i ≥ 0; i--) {
            while (s.size() && s.top() ≤ a[i]) {
                s.pop();
            }
            if (s.size() && s.top() > a[i]) {
                ans[i] = s.top();
            } else {
                ans[i] = -1;
            }
            s.push(a[i]);
        }
        for (int i = 0; i < n; i++) {
            nums[i] = ans[i];
        }
        return nums;
    }
};
```



**THANK  
YOU !**

