

Valid Paranthesis

str = "({})"

str = "({ })"

str = "({})"

Valid ({ }) Strings / Paranthesis

→ Opening bracket must have some closing bracket.

→ Order must be same for opening & closing

→ Equal number of opening & closing

({ { } }) [] last opening bracket → first closing bracket

→ LIFO fits here

```
bool isValid(str) {
```

```
    stack<char> st;
```

```
    for (i = 0 to n) {
```

```
        if (str[i] == opening)
```

```
            st.push(ch)
```

TC = $O(n)$

SC = $O(n)$

```
    else
```

```
        if (st.size() == 0)
```

```
            return false
```

closing > opening

```
        if (str[i] match with st.top())
```

```
            st.pop()
```

```
    else return false
```

```
    }
```

```
    return st.size() == 0
```

opening > closing