Preorder Traversals

Prof. Harish D.G.
Dept. of Computer and IT
College of Engineering,Pune(COEP)
www.harishgadade.com

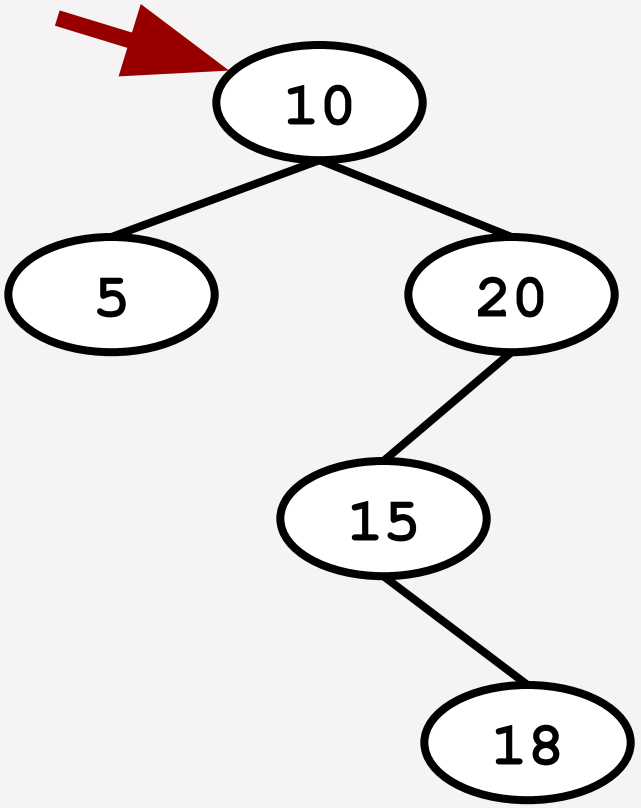
Binary Tree Traversals

1. Binary Tree Traversals
   - Inorder Traversal
   - Preorder Traversal
   - Postorder Traversal

Preorder Algorithm (VLR)

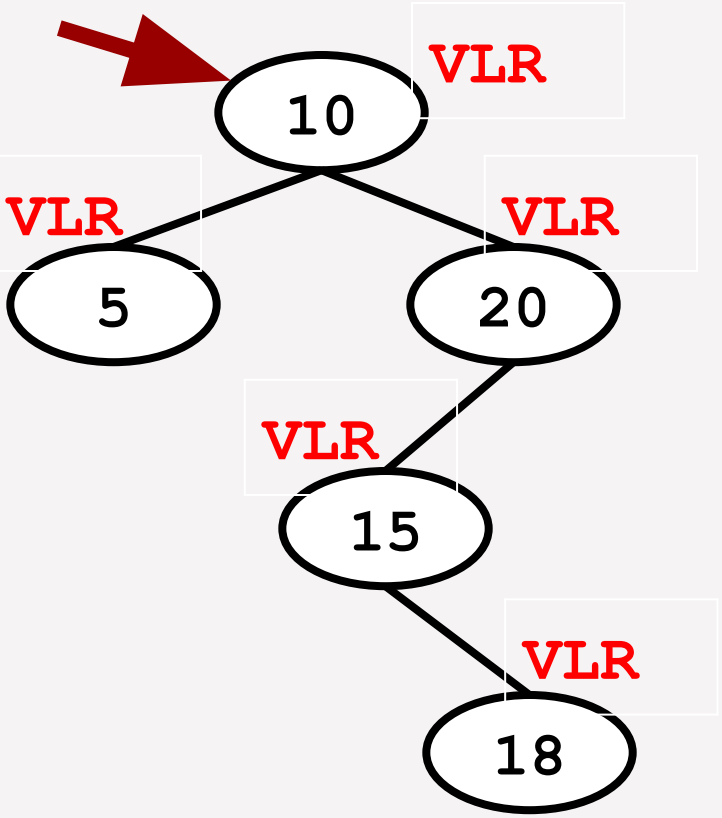
1. Visit the root
2. Traverse the Left sub-tree in preorder
3. Traverse the Right sub-tree in preorder

2. Preorder Traversals (VLR)



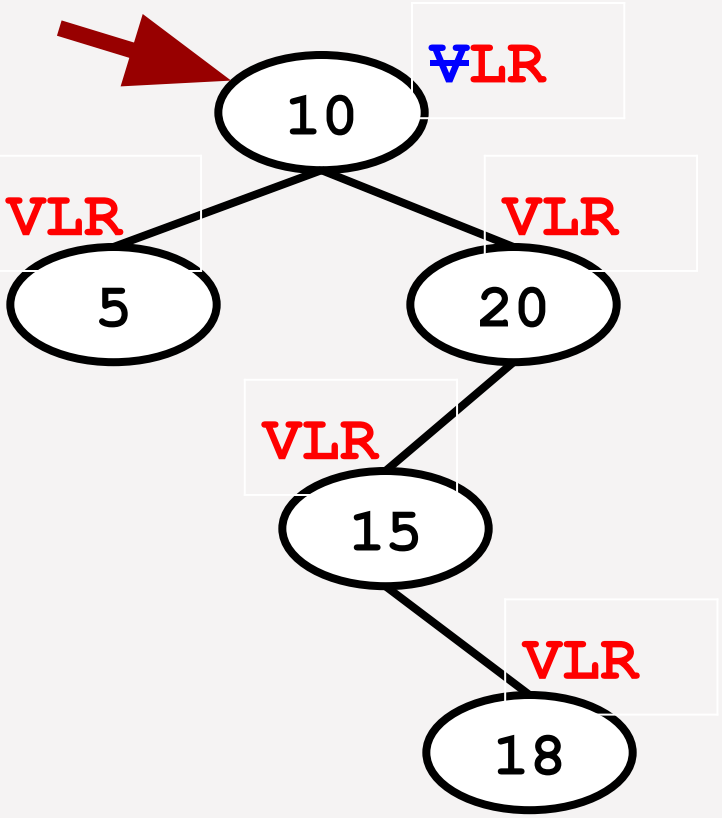

```
Output :
```

2. Preorder Traversals (VLR)



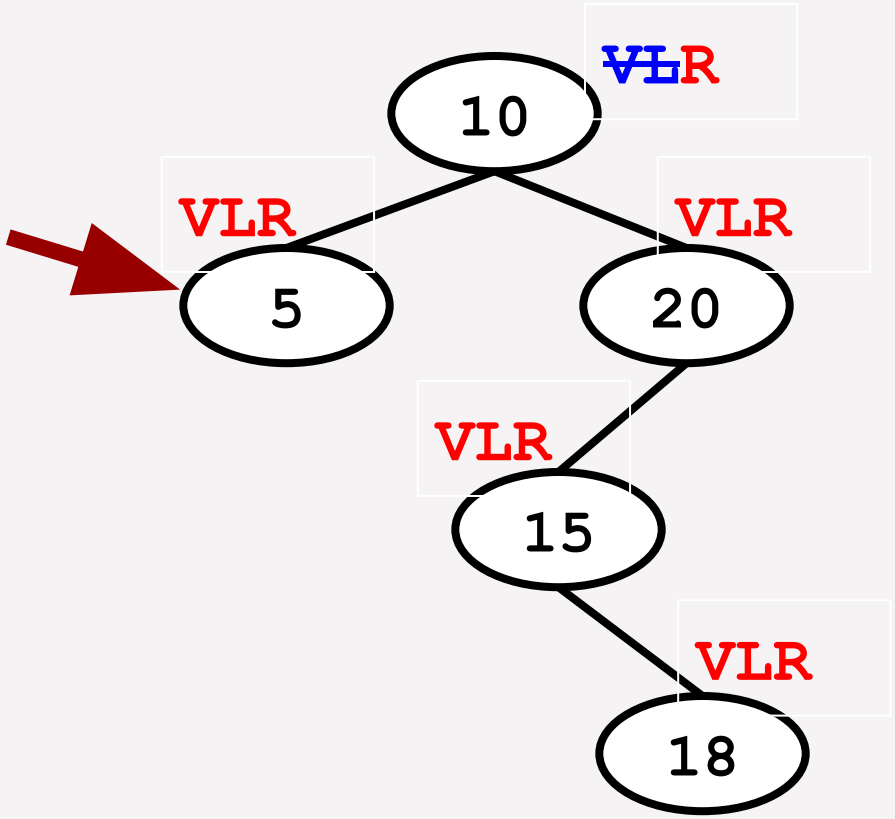

```
Output :
```

2. Preorder Traversals (VLR)



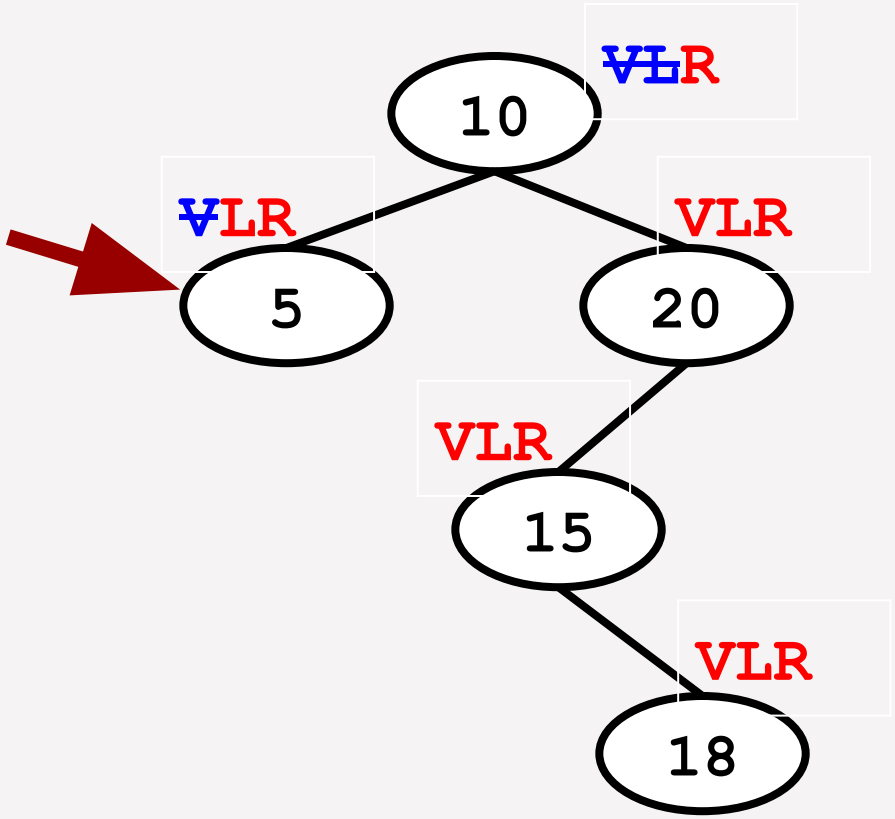

```
Output : 10
```

2. Preorder Traversals (VLR)



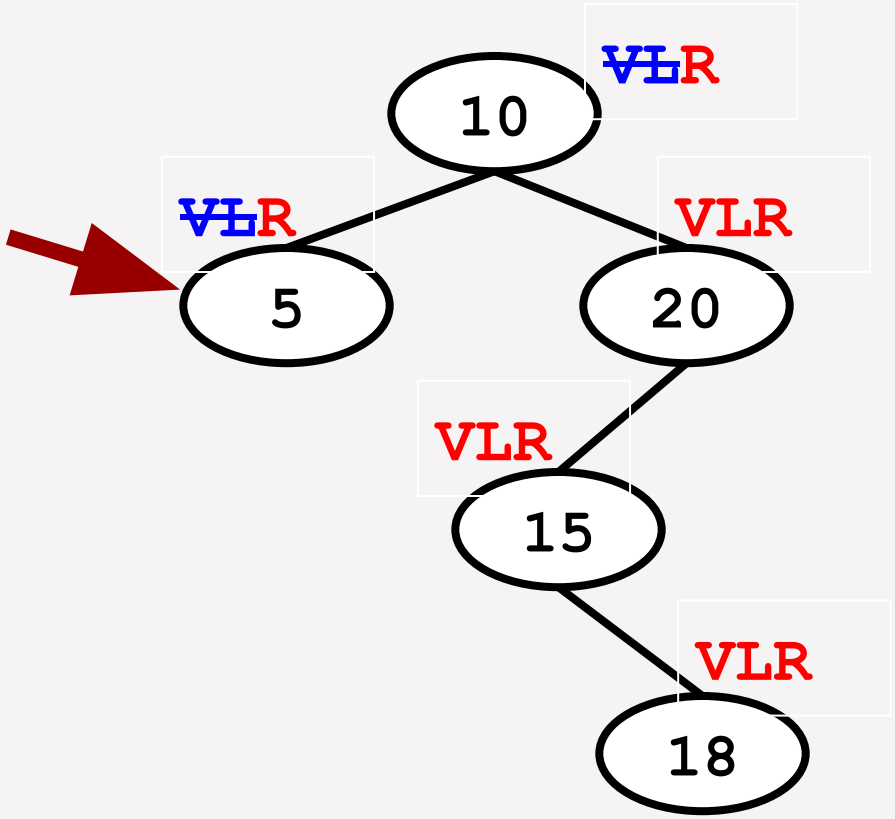

```
Output : 10
```

2. Preorder Traversals (VLR)



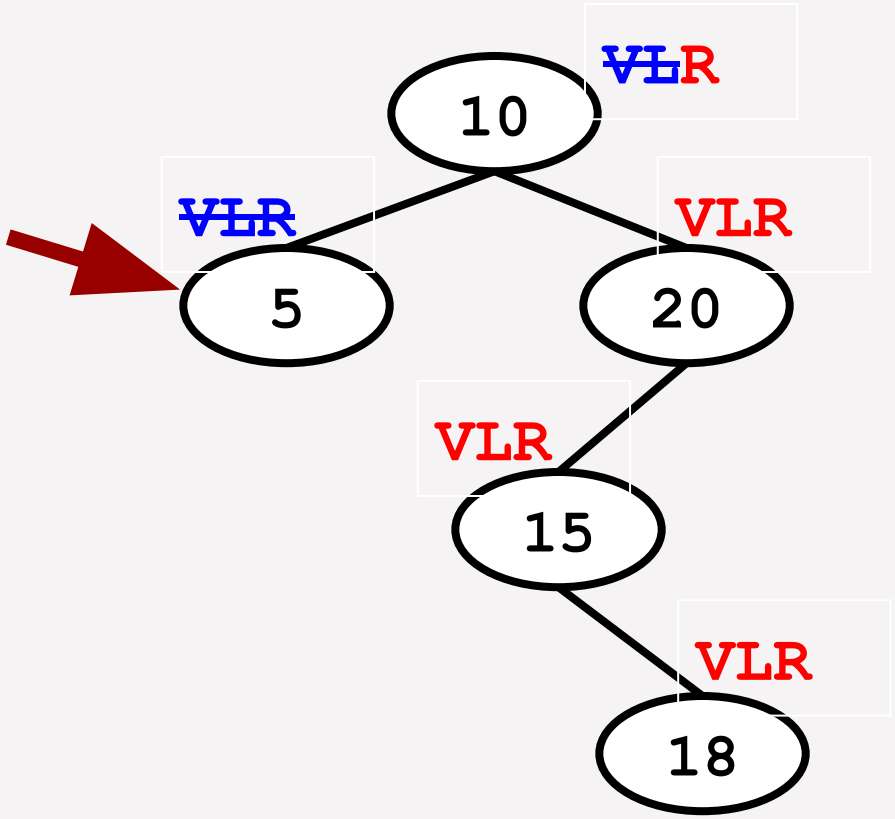

```
Output : 10, 5
```

2. Preorder Traversals (VLR)



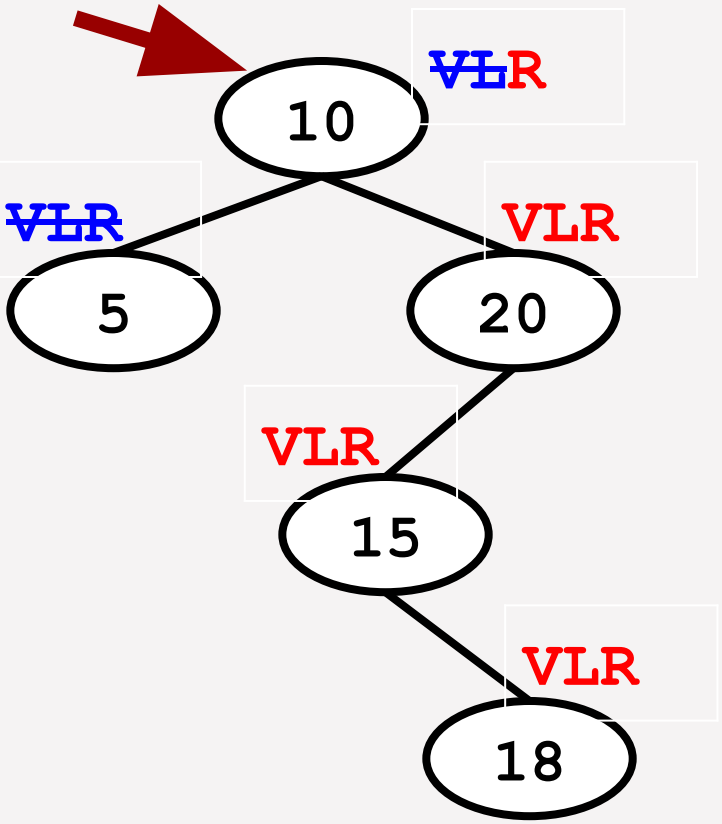

```
Output : 10, 5
```

2. Preorder Traversals (VLR)



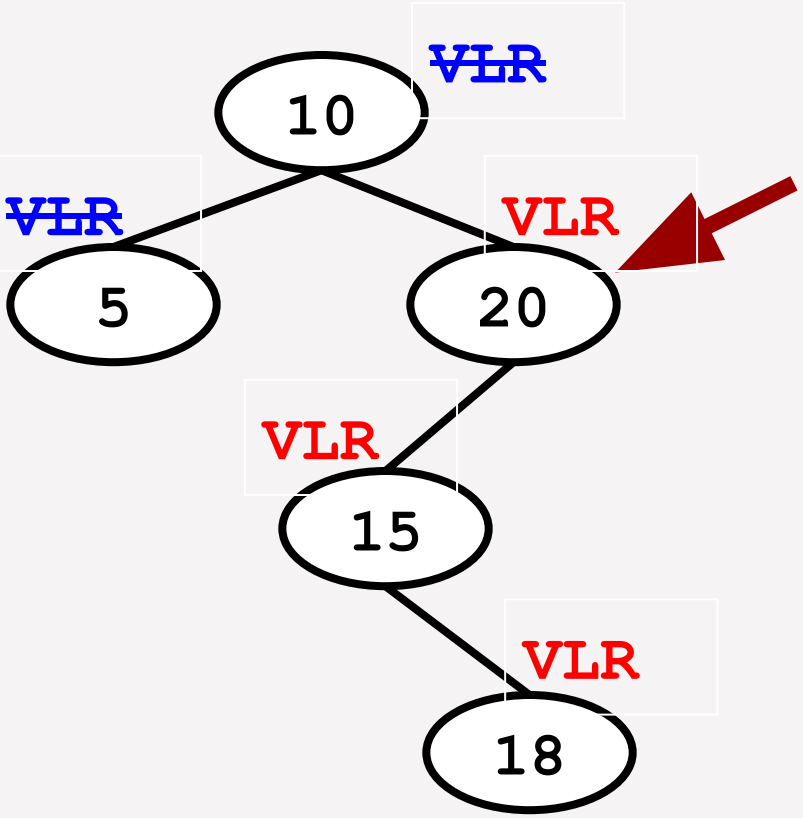

```
Output : 10, 5
```

2. Preorder Traversals (VLR)



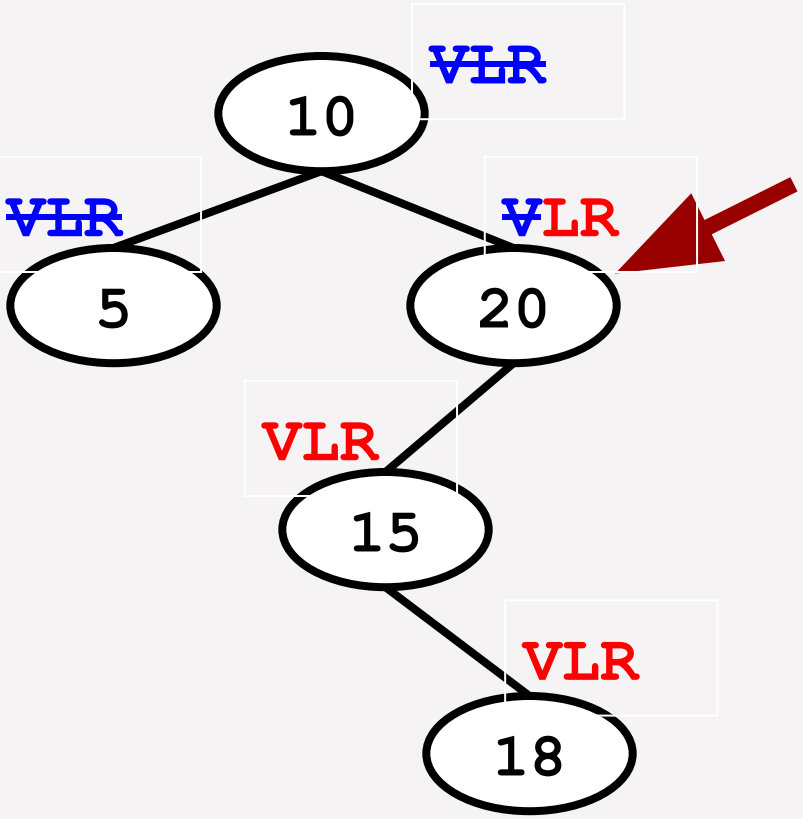

```
Output : 10, 5
```

2. Preorder Traversals (VLR)



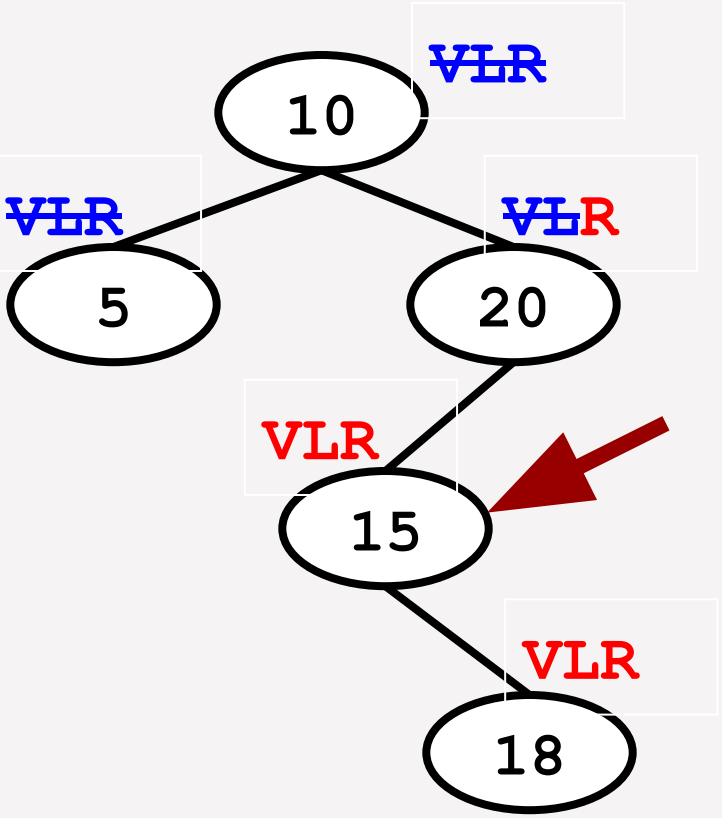

```
Output : 10, 5
```

2. Preorder Traversals (VLR)



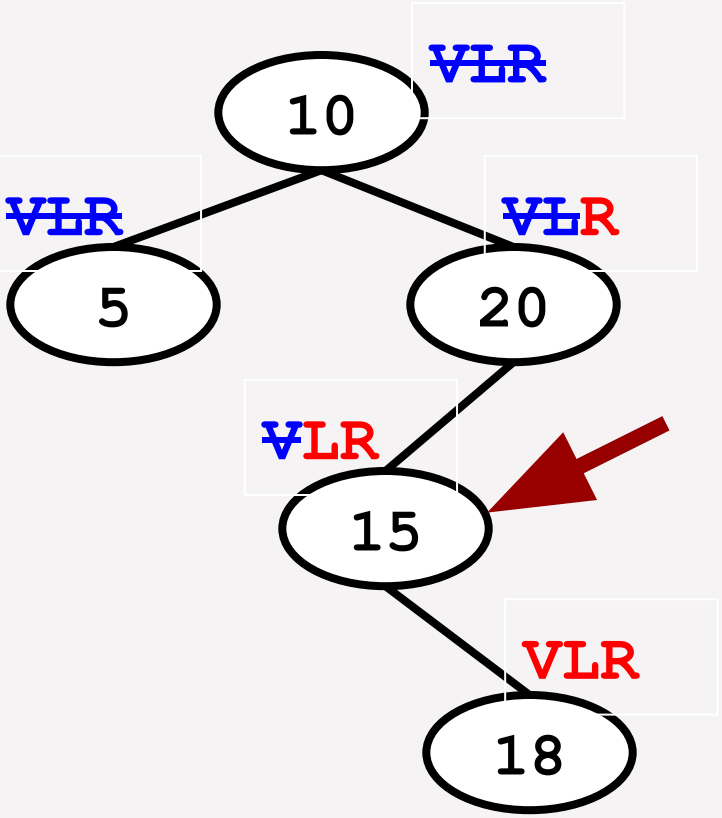

```
Output : 10, 5, 20
```

2. Preorder Traversals (VLR)



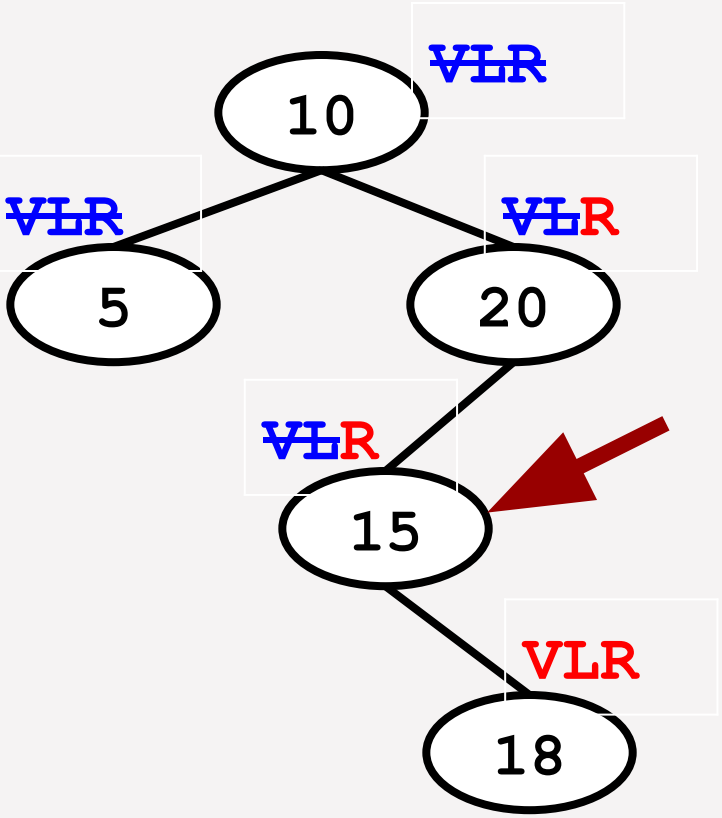

```
Output : 10, 5, 20
```

2. Preorder Traversals (VLR)



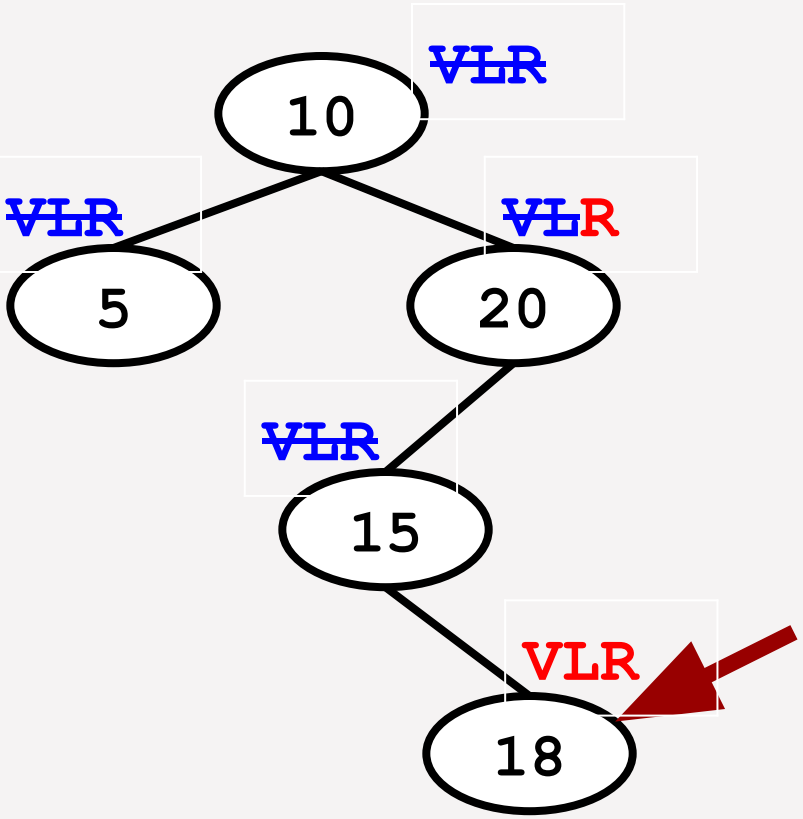

```
Output : 10, 5, 20, 15
```

2. Preorder Traversals (VLR)



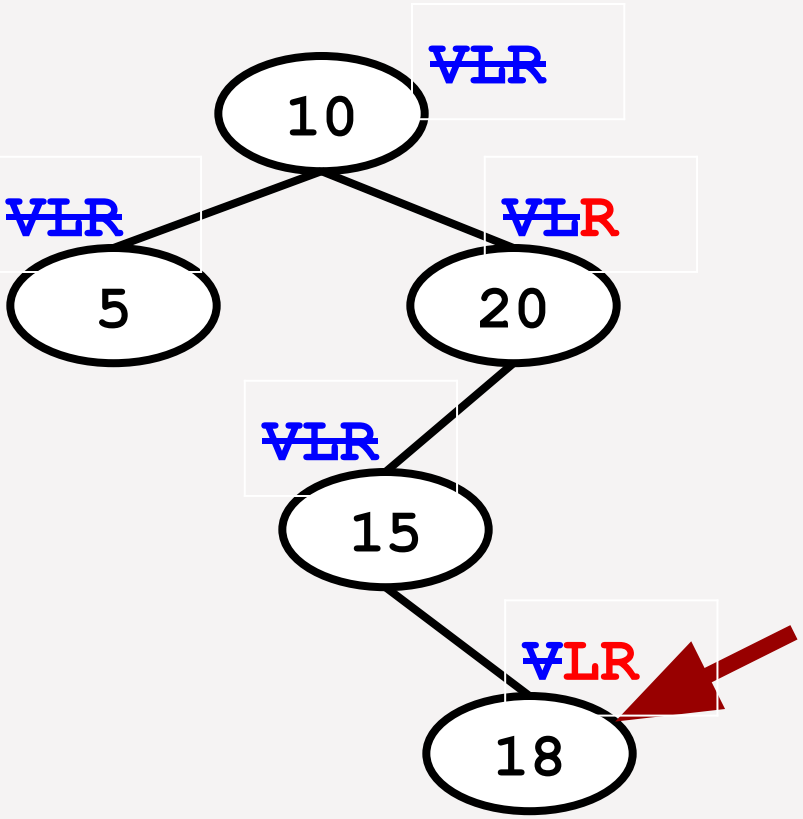

```
Output : 10, 5, 20, 15
```

2. Preorder Traversals (VLR)



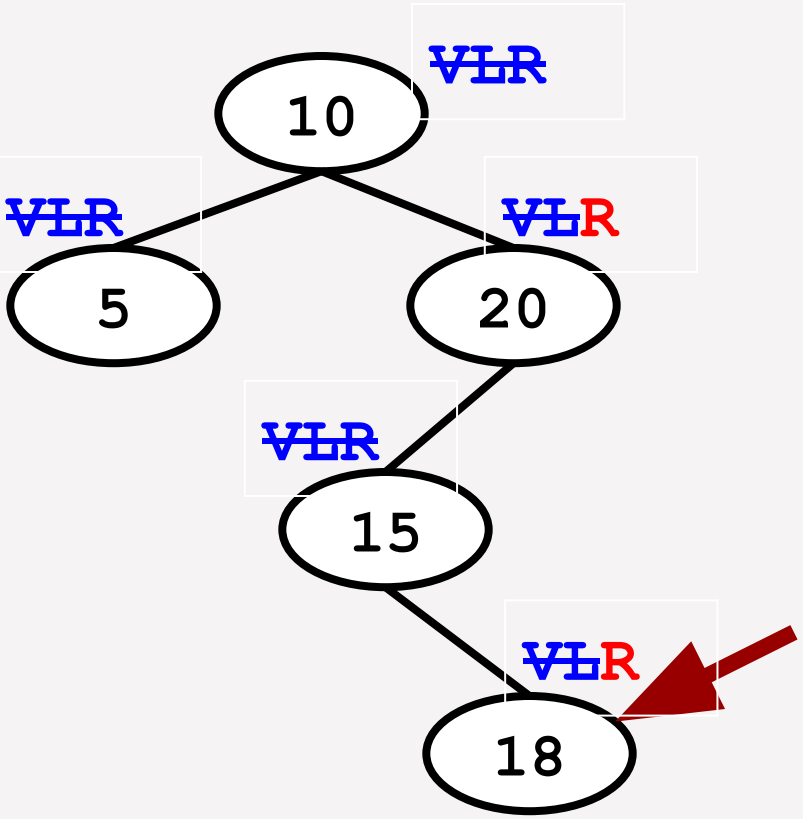

```
Output : 10, 5, 20, 15
```

2. Preorder Traversals (VLR)



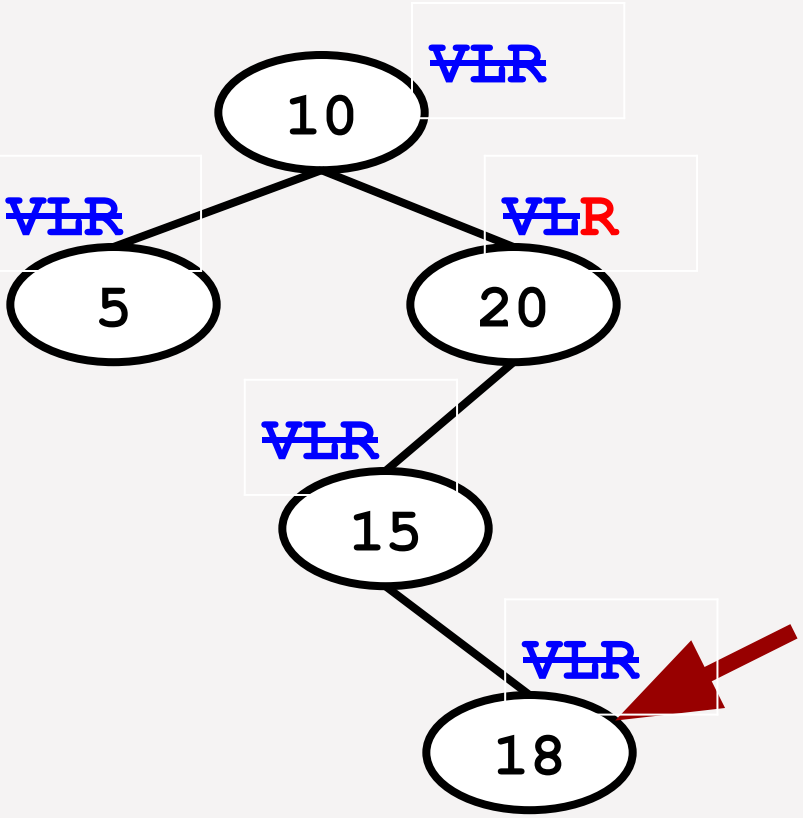

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)



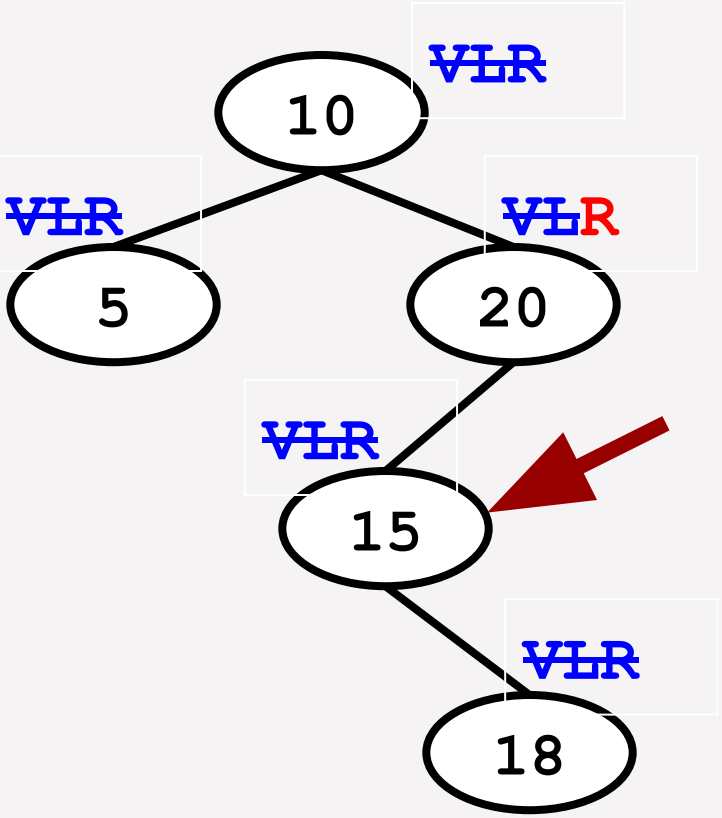

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)



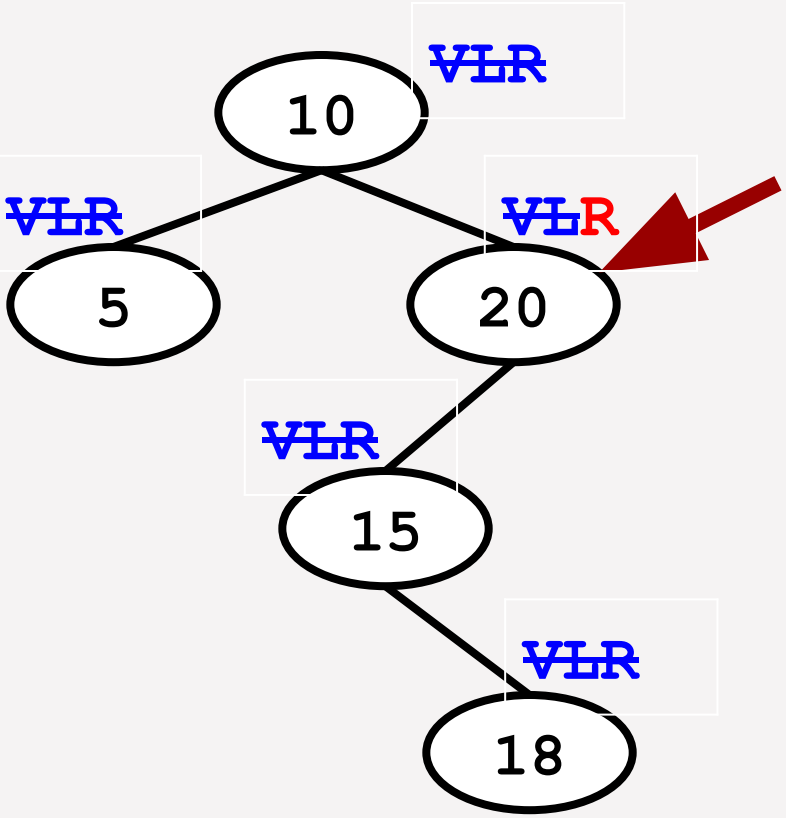

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)



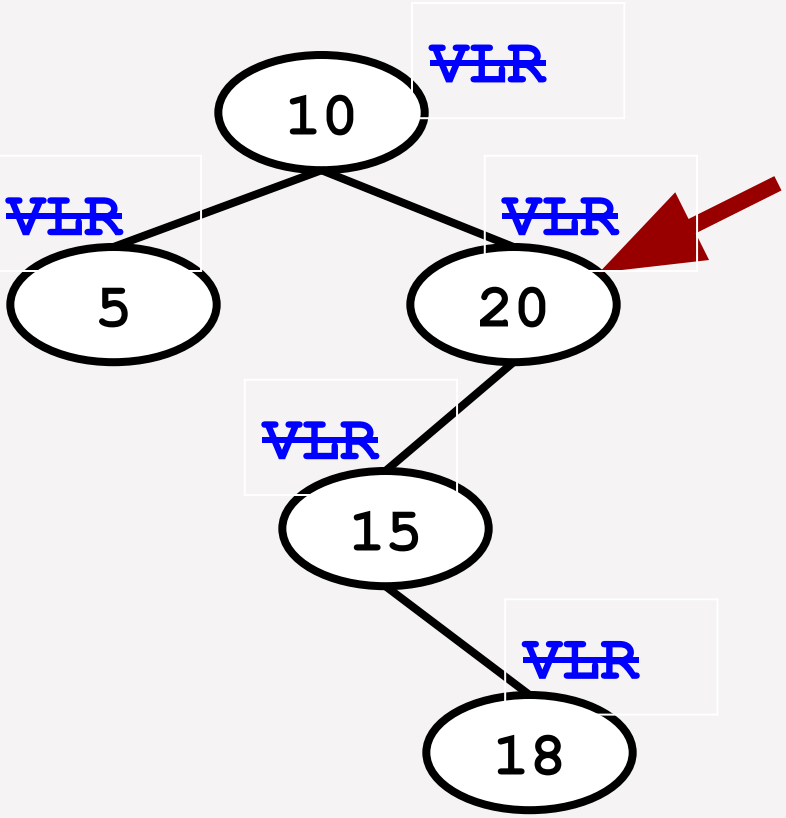

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)



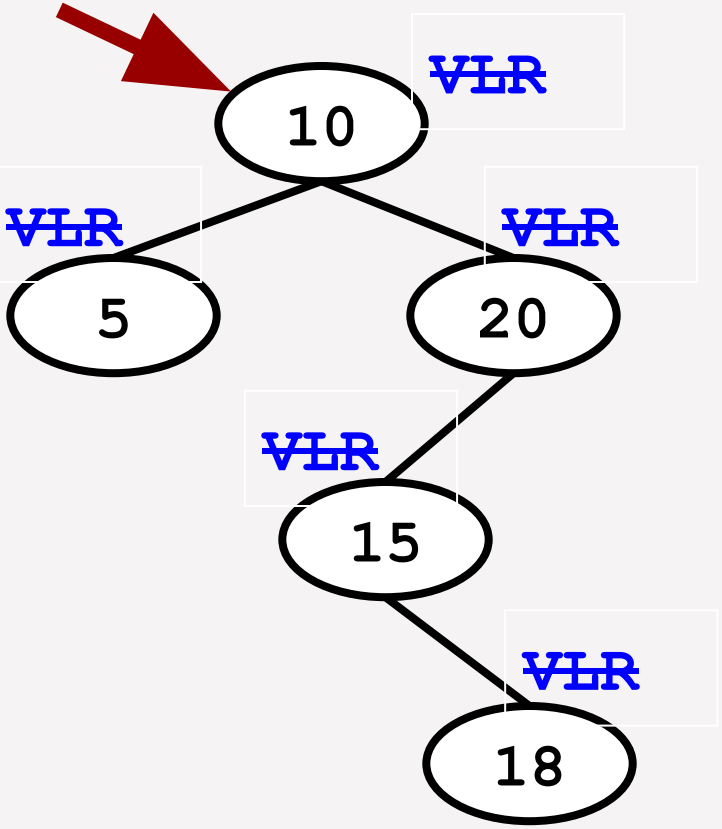

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)



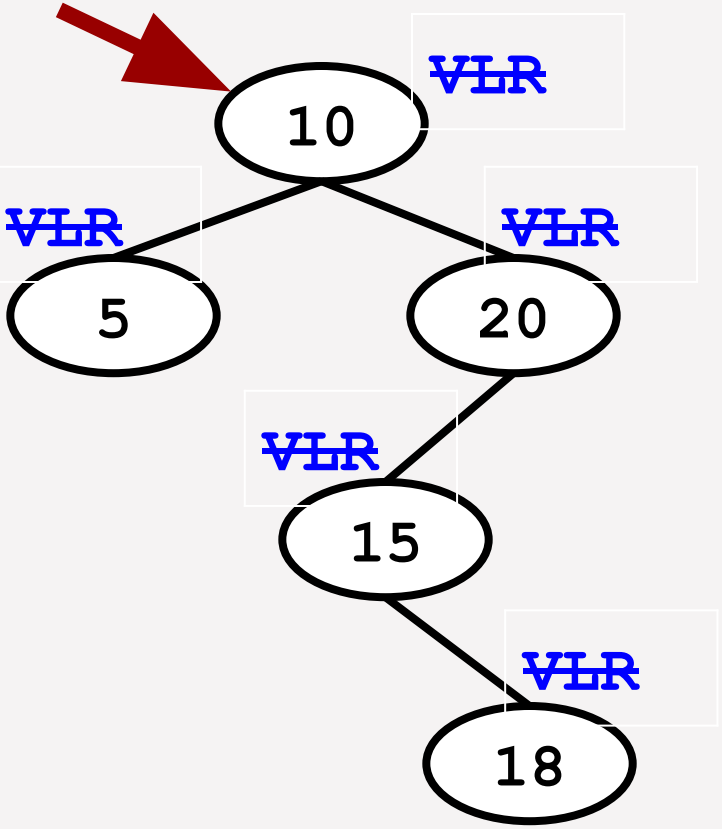

```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)




```
Output : 10, 5, 20, 15, 18
```

2. Preorder Traversals (VLR)




```
Output : 10, 5, 20, 15, 18
                    Final Preorder Traversals
```