

## Activity 2 – Stack Programs

### Infix to Postfix Conversion

```
import java.util.Stack;
import java.util.Scanner;

public class InfixToPostfix {

    static int precedence(char op) {
        switch (op) {
            case '+':
            case '-': return 1;
            case '*':
            case '/': return 2;
            case '^': return 3;
        }
        return -1;
    }

    public static String convert(String exp) {
        Stack<Character> st = new Stack<>();
        String result = "";

        for (char ch : exp.toCharArray()) {

            if (Character.isLetterOrDigit(ch))
                result += ch;

            else if (ch == '(')
                st.push(ch);

            else if (ch == ')') {
                while (!st.isEmpty() && st.peek() != '(')
                    result += st.pop();
                st.pop();
            }

            else {
                while (!st.isEmpty() && precedence(st.peek()) >= precedence(ch))
                    result += st.pop();
                st.push(ch);
            }
        }

        while (!st.isEmpty())
            result += st.pop();

        return result;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter infix: ");
        String exp = sc.nextLine();

        System.out.println("Postfix: " + convert(exp));
    }
}
```

### Reverse Array Using Stack

```
import java.util.Stack;

public class ReverseArrayUsingStack {
    public static void main(String[] args) {
```

```
int[] arr = {1,2,3,4,5};
Stack<Integer> st = new Stack<>();

for(int num : arr)
    st.push(num);

for(int i=0;i<arr.length;i++)
    arr[i] = st.pop();

System.out.print("Reversed: ");
for(int num : arr)
    System.out.print(num + " ");
}
```