

Job No: 07

Job Name: Write a program to convert expression from postfix to infix

Theory:

When the operator is written in between the operands, then it is known as infix notation. For example, $(p + q) * (r + s)$.

The postfix expression is an expression in which the operator is written after the operands. For example, the postfix expression of infix notation $(2+3)$ can be written as $23+$.

Code:

```
function detectOperand(ch) {
    return (ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z');
}

function getInfix(postfix) {
    let stack = [];

    for (let value of postfix){
        if(detectOperand(value)){
            stack.push(value);
        } else {
            let firstOperand = stack.pop();
            let secondOperand = stack.pop();
            stack.push("(" + secondOperand + value + firstOperand + ")");
        }
    }

    return stack[stack.length-1];
}

let postfix = "abc*d+";
console.log(`The given postfix expression is: ${postfix}`)
console.log(`The Infix expression is: ${getInfix(postfix)}`);
```

Input/Output:

```
The given postfix expression is: abc*d+
The Infix expression is: ((a+(b*c))+d)
```

Job No: 07

Job Name: Write a program to convert expression from postfix to infix

Theory:

When the operator is written in between the operands, then it is known as infix notation. For example, $(p + q) * (r + s)$.

The postfix expression is an expression in which the operator is written after the operands. For example, the postfix expression of infix notation $(2+3)$ can be written as $23+$.

Code:

```
const findOperand =(x) => ((x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z'));

const takeInfix =(input) => {
  let stack = [];

  for (let i = 0; i < input.length; i++)
  {
    if(findOperand(input[i])){
      stack.push(input[i]);
    } else {
      let topFirstElement = stack.pop();
      let topSecondElement = stack.pop();

      stack.push("(" + topSecondElement + input[i] +
        topFirstElement + ")");
    }
  }
  return stack[stack.length-1];
}

let input = "AB/CD*+";
console.log(`The postfix expression is: ${input}`)
console.log(`The Infix expression is: ${takeInfix(input)}`);
```

Input/Output:

```
The given postfix expression is: AB/CD*+
The Infix expression is: ((A/B)+(C*D))
```