

```

public class Notacao {

    Stack pilha;
    public Notacao() {
        this.pilha = new Stack();
    }

    public String infixaParaPosfixa(String infixa) {

        char digito;
        String posfixa = new String("");

        for (int i = 0; i <= infixa.length() - 1; i++ ) {
            digito = infixa.charAt(i);
            if( pilha.isEmpty()) {
                if(eOperador(digito) || digito == '(') {
                    pilha.push(digito + "");
                } else {
                    posfixa += digito;
                }
            } else {
                if(digito == '(') {
                    pilha.push(digito + "");
                }
                else if(eOperador(digito)) {
                    if(precedenciaDeOperador(pilha.peek()) < precedencia
DeOperador(digito + "")) {
                        pilha.push(digito + "");
                    } else if(precedenciaDeOperador(pilha.peek()) > prece
denciaDeOperador(digito + "") ||
                        precedenciaDeOperador(pilha.peek()) == prec
edenciaDeOperador(digito + "")) {

                        while (precedenciaDeOperador(pilha.peek()) > prec
edenciaDeOperador(digito + "") ||
                            precedenciaDeOperador(pilha.peek()) == precedenci
aDeOperador(digito + "")) {
                            posfixa += pilha.pop();
                        }
                        pilha.push(digito + "");
                    }
                } else if (digito == ')'){
                    while(!pilha.peek().equals("(")) {
                        posfixa += pilha.pop();
                    }
                    pilha.pop();
                } else {
                    posfixa += digito;
                }
            }
        }
    }
}

```

```

    }
}

while (!pilha.isEmpty()) {
    posfixa += pilha.pop();
}

return posfixa;
}

public String infixaParaPrefixa(String infixa) {
    String infixaAux = new String("");
    String prefixa = new String("");
    char digito;

    // Invertendo a String
    for(int z = infixa.length() - 1; z >=0; z--) {
        infixaAux += infixa.charAt(z);
    }
    for(int x = 0; x <= infixa.length() -1; x++) {
        digito = infixaAux.charAt(x);

        if( pilha.isEmpty()) {

            if(eOperador(digito) || digito == ')') {
                pilha.push(digito + "");
            } else {
                prefixa += digito;
            }

        } else {

            if(digito == ')') {
                pilha.push(digito + "");
            }

            else if(eOperador(digito)) {
                if(precedenciaDeOperador(digito + "") > precedenciaDeOperador(pilha.peek()) || pilha.peek().equals("")) || precedenciaDeOperador(digito + "") == precedenciaDeOperador(pilha.peek())) {
                    pilha.push(digito + "");
                } else {
                    while (precedenciaDeOperador(digito + "") < precedenciaDeOperador(pilha.peek())) {
                        prefixa += pilha.pop();
                    }
                    pilha.push(digito + "");
                }
            }
        }
    }
}

```

```

        } else if (digito == '(') {
            while(!pilha.peek().equals("")) {
                prefixa += pilha.pop();
            }
            pilha.pop();
        } else {
            prefixa += digito;
        }
    }
}

while (!pilha.isEmpty()) {
    prefixa += pilha.pop();
}

infixaAux = "";
for(int z = prefixa.length() - 1; z >=0; z--) {
    infixaAux+= prefixa.charAt(z);
}
prefixa = infixaAux;
return prefixa;
}

public String posfixaParaInfixa(String posfixa) {
    String infixa = new String("");
    char digito;

    for (int c = 0; c < posfixa.length(); c++) {
        digito = posfixa.charAt(c);

        if(!eOperador(digito)) {
            pilha.push(digito + "");
        } else {
            String b = pilha.pop();
            String a = pilha.pop();

            infixa = "(" + a + digito + b + " ";
            pilha.push(infixa);
        }
    }
    infixa=pilha.pop();
    // while (!pilha.isEmpty()) {
    //     infixa += pilha.pop();
    // }
    return infixa;
}

public String prefixaParaInfixa(String prefixa) {
    String infixa = new String("");

```

```

char digito;

for (int c = prefixa.length() - 1; c >= 0; c--) {
    digito = prefixa.charAt(c);

    if(!eOperador(digito)) {
        pilha.push(digito + "");
    } else {
        String a = pilha.pop();
        String b = pilha.pop();

        infix = "(" + a + digito + b + ")";
        pilha.push(infix);
    }
}
infixa="";
while (!pilha.isEmpty()) {
    infix += pilha.pop();
}
return infix;
}

```

```

public boolean eOperador(char digito) {
    switch(digito) {
        case '+':
        case '-':
        case 'x':
        case 'X':
        case '*':
        case '/':
            return true;
    }
    return false;
}

```

```

public int precedenciaDeOperador(String operador){
    switch(operador) {
        case "+":
        case "-":
            return 1;
        case "x":
        case "X":
        case "/":
        case "*":
            return 2;
    }
    return 0;
}

```

```
}  
  
}
```