

**SSN COLLEGE OF ENGINEERING`
DEPARTMENT OF CSE**

DATA STRUCTURES LABORATORY (CS8381)

Ex. No. 5 Evaluation of Expression using Array Implementation of Stack ADT

i) Create a stack ADT using arrays and perform the following operations:

- a. Push
- b. Pop
- c. Returning top of the stack
- d. Check whether stack is full
- e. Check whether stack is empty
- f. Disposing stack
- g. Display stack

Use the following structure to create the stack ADT:

```
struct stackrecord
{
    int capacity;
    int topofstack;
    elementtype *array;
};
```

ii) Use the above implementation of stack ADT to convert infix expression to postfix and evaluate the same by assigning values to the operands. Use character stack for conversion and integer stack for evaluation.

Test cases:

- 1) Infix expression: $a + b * c$
Postfix expression: $abc*+$
- 2) Infix expression: $(a+b*c-d)/(e*f)$
Postfix expression: $abc*+d-ef*/$
- 3) Infix expression: $(a+b) * (c+d) - e$
Postfix expression: $ab+cd+*e-$

Note: No global declaration should be made. The main program should have the conversion and evaluation functions and they should make use of the stack ADT.