**Aim:** Convert an Infix expression to Postfix expression using stack ADT.

#include <stdio.h> #include <stdlib.h> #include <ctype.h> #include <string.h> int prec(char c) {

if (c == '^') return 3;

else if (c == '/' || c == '\*') return 2;

else if (c == '+' || c == '-') return 1;

else return-1;

}

void infixToPostfix(char\* exp) {

int len = strlen(exp); char result[len + 1]; char stack[len];

int j = 0; int top =-1;

for (int i = 0; i < len; i++) { char c = exp[i];

if (isalnum(c)) result[j++] = c;

else if (c == '(')

stack[++top] = '(';

else if (c == ')') {

while (top !=-1 && stack[top] != '(') { result[j++] = stack[top--];

}

top--;

}

else {

while (top !=-1 && (prec(c) < prec(stack[top]) || prec(c) == prec(stack[top]))) {

result[j++] = stack[top--];

}

stack[++top] = c;

}

}

while (top !=-1) {

result[j++] = stack[top--];

}

result[j] = '\0'; printf("%s\n", result);

}

int main() {

char exp[] = "a+b\*(c^d-e)^(f+g\*h)-i"; infixToPostfix(exp);

return 0;

}