15. Write a C Program to implement the operator precedence parsing.

CODE:

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

#define MAX 100

char stack[MAX];

int top = -1;

void push(char c) {

if (top >= MAX - 1) {

printf("Stack overflow\n");

exit(1);

}

stack[++top] = c;

}

char pop() {

if (top == -1) {

printf("Stack underflow\n");

exit(1);

}

return stack[top--];

}

char peek() {

if (top == -1) {

return '\0';

}

return stack[top];

}

int precedence(char op) {

switch (op) {

case '+':

case '-':

return 1;

case '\*':

case '/':

return 2;

default:

return 0;

}

}

int isOperator(char c) {

return c == '+' || c == '-' || c == '\*' || c == '/';

}

void parse(char \*expr) {

for (int i = 0; expr[i] != '\0'; i++) {

if (isdigit(expr[i])) {

printf("%c ", expr[i]);

} else if (isOperator(expr[i])) {

while (top != -1 && precedence(peek()) >= precedence(expr[i])) {

printf("%c ", pop());

}

push(expr[i]);

} else {

printf("Invalid character\n");

exit(1);

}

}

while (top != -1) {

printf("%c ", pop());

}

printf("\n");

}

int main() {

char expr[] = "3+5\*2-8"; // Example input expression

parse(expr);

return 0;

}

