15.

#include <stdio.h>

#include <ctype.h>

char lookahead;

void match(char expected) {

if (lookahead == expected) lookahead = getchar();

else { printf("Syntax Error\n"); exit(1); }

}

void expr();

void term();

void factor();

void expr() {

term();

while (lookahead == '+' || lookahead == '-') {

match(lookahead); // Handle '+' or '-'

term();

}

}

void term() {

factor();

while (lookahead == '\*' || lookahead == '/') {

match(lookahead); // Handle '\*' or '/'

factor();

}

}

void factor() {

if (isdigit(lookahead)) {

while (isdigit(lookahead)) match(lookahead); // Consume digits

} else if (lookahead == '(') {

match('(');

expr();

match(')');

} else { printf("Syntax Error\n"); exit(1); }

}

int main() {

lookahead = getchar(); // Get first character

expr();

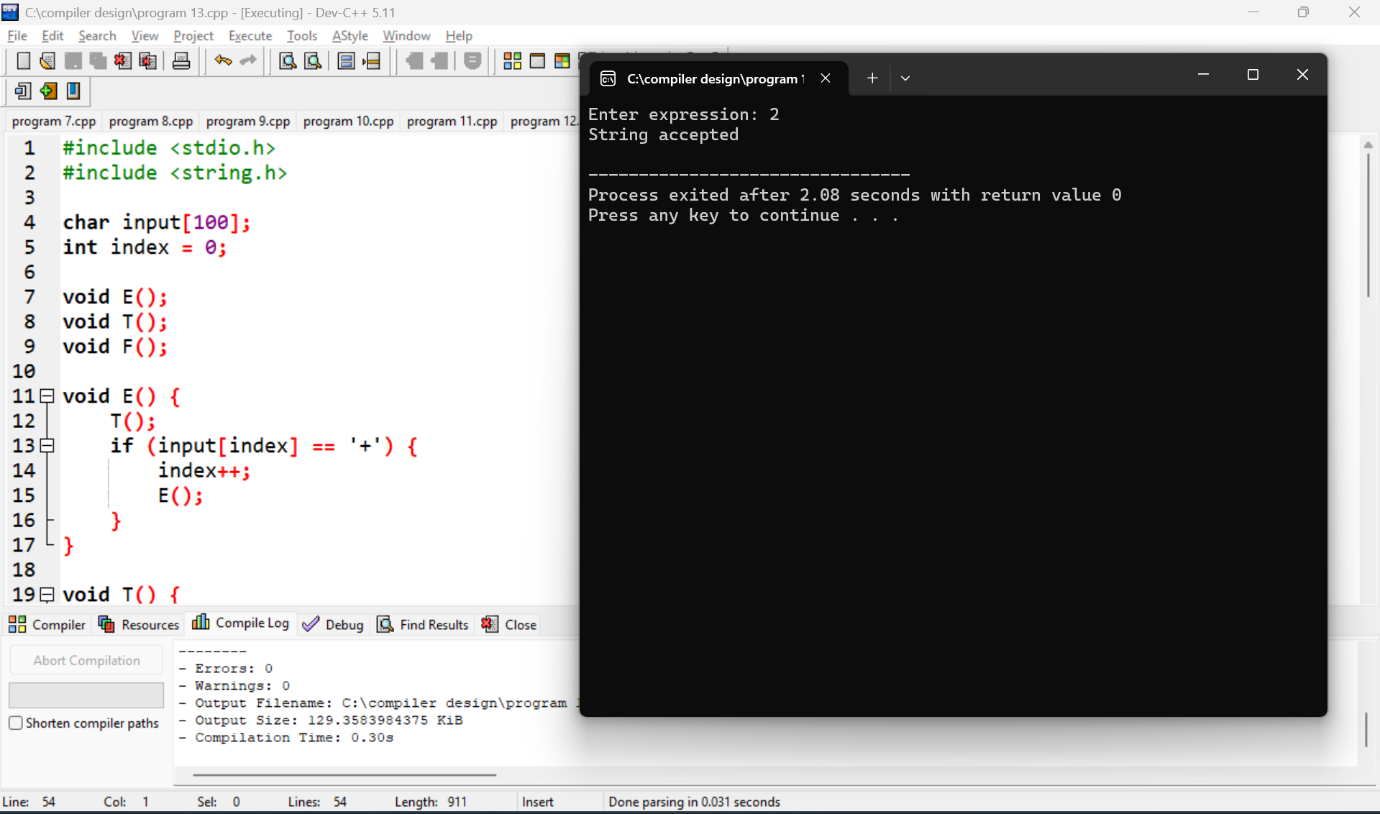
if (lookahead == '\n' || lookahead == EOF) printf("Parsing successful!\n");

else printf("Syntax Error\n");

return 0;

}

**Output:**

****

16.

#include <stdio.h>

int temp\_count = 1;

void generate\_three\_address\_code(char \*op, char \*op1, char \*op2) {

printf("t%d = %s %s %s\n", temp\_count++, op1, op, op2);

}

int main() {

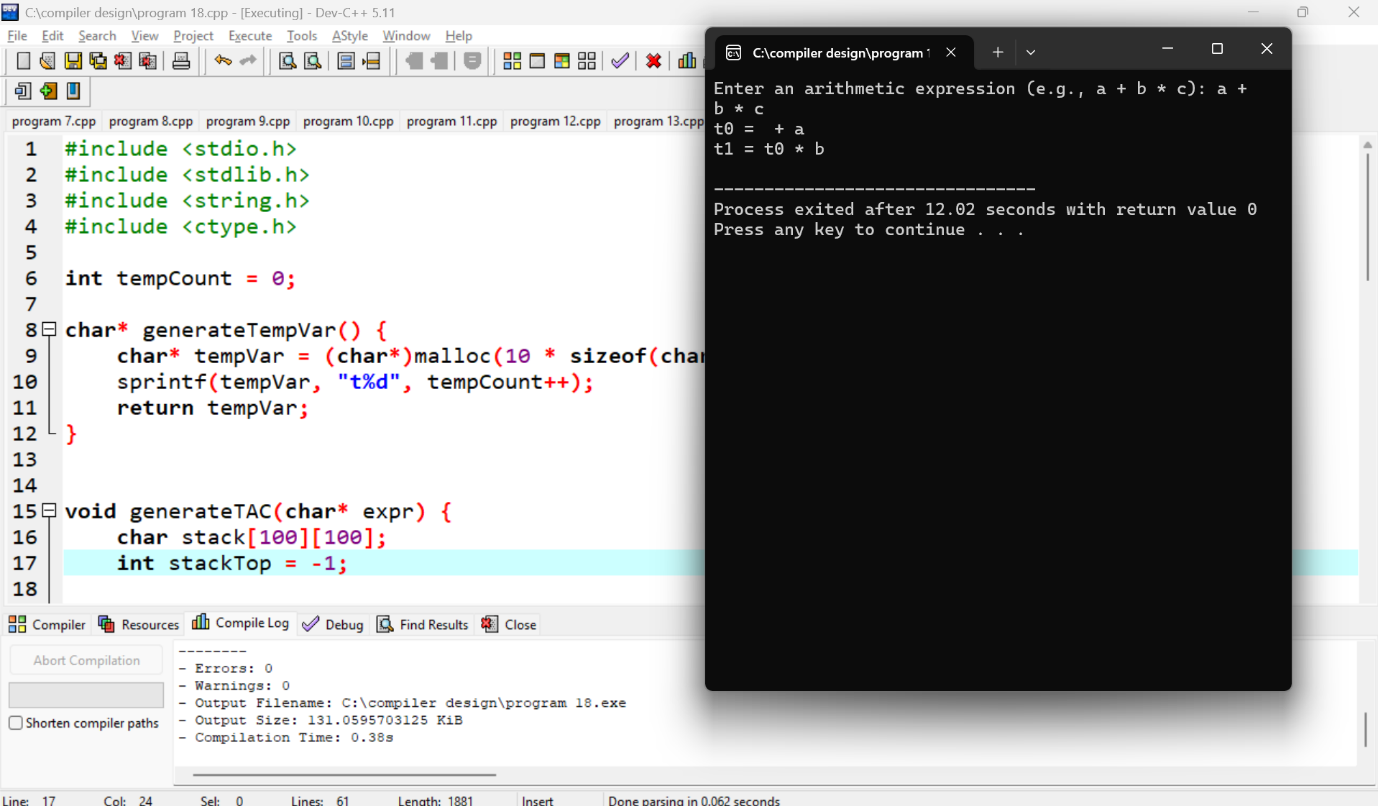
generate\_three\_address\_code("+", "a", "b");

generate\_three\_address\_code("\*", "t1", "c");

return 0;

}

**Output:**

****