**10. Write a C program to construct recursive descent parsing.**

**Code:**

**#include <stdio.h>**

**#include <ctype.h>**

**#include <string.h>**

**#include <stdlib.h>**

**#define MAX\_LEN 100**

**const char \*input;**

**int pos = 0;**

**void expr();**

**void expr\_prime();**

**void term();**

**void term\_prime();**

**void factor();**

**void error();**

**void match(char expected) {**

**if (input[pos] == expected) {**

**pos++;**

**} else {**

**error();**

**}**

**}**

**void error() {**

**printf("Syntax Error at position %d\n", pos);**

**exit(1);**

**}**

**void expr() {**

**term();**

**expr\_prime();**

**}**

**void expr\_prime() {**

**if (input[pos] == '+') {**

**match('+');**

**term();**

**expr\_prime();**

**}**

**}**

**void term() {**

**factor();**

**term\_prime();**

**}**

**void term\_prime() {**

**if (input[pos] == '\*') {**

**match('\*');**

**factor();**

**term\_prime();**

**}**

**}**

**void factor() {**

**if (isdigit(input[pos])) {**

**while (isdigit(input[pos])) {**

**pos++;**

**}**

**} else if (input[pos] == '(') {**

**match('(');**

**expr();**

**match(')');**

**} else {**

**error();**

**}**

**}**

**int main() {**

**char str[MAX\_LEN];**

**printf("Enter an arithmetic expression: ");**

**scanf("%s", str);**

**input = str;**

**expr();**

**if (input[pos] == '\0') {**

**printf("Parsing Successful!\n");**

**} else {**

**error();**

**}**

**return 0;**

**}**

