```
// RECURSIVE DESCENT PARSING

#include<stdio.h>
#include<ctype.h>
#include<string.h>
void Tprime();
void Eprime();
void E();
void check();
void T();
char expression[10];
int count, flag;
int main()
{
        count = 0;
        flag = 0;
        printf("\nEnter an Algebraic Expression:\t");
        scanf("%s", expression);
        E();
        if((strlen(expression) == count) && (flag == 0))
                printf("\nThe Expression %s is Valid\n", expression);
        else
                printf("\nThe Expression %s is Invalid\n", expression);
}
void E()
{
        T();
        Eprime();
}
```

```
void T()
{
        check();
        Tprime();
}
void Tprime()
{
        if(expression[count] == '*')
        {
                count++;
                check();
                Tprime();
        }
}
void check()
{
        if(isalnum(expression[count]))
        {
                count++;
        }
        else if(expression[count] == '(')
        {
                count++;
                E();
                if(expression[count] == ')')
                        count++;
                else
                        flag = 1;
        }
```

```
    else
    {
        flag = 1;
    }
}
void Eprime()
{
    if(expression[count] == '+')
    {
        count++;
        T();
        Eprime();
    }
}
```