Name : Rajvardhan Reddy
Reg No : 180905093
Sec : B
Roll No : 19

CD LAB - 6 : RECURSIVE DESCENT PARSER

Example problem :
E --> TEprime
E' --> +TE' /
T --> FT'
T' --> *FT' /
F --> (E) / i

program :
```
/*
E --> TEprime
E' --> +TE' /
T --> FT'
T' --> *FT' /
F --> (E) / i
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int curr = 0;
char str[100];
//
--------------------------------------------------------------------------------------------------
----------------
void E();
void Eprime();
void T();
void Tprime();
void F();
//
---------------------------------------------------------------------------------------------------
void invalid()
{
printf("-----------------ERROR!----------------\n");
exit(0);
}
```

```c
void valid()
{
printf("----------------SUCCESS!---------------\n");
exit(0);
}
//
----------------------------------------------------------------------------------------------------
----------------
void E()
{
T();
Eprime();
}
//
----------------------------------------------------------------------------------------------------
----------------
void Eprime()
{
if(str[curr] == '+')
{
curr++;
T();
Eprime();
}
}
//
----------------------------------------------------------------------------------------------------
----------------
void T()
{
F();
Tprime();
}
//
----------------------------------------------------------------------------------------------------
----------------
void Tprime()
{
if(str[curr] == '*')
{
curr++;
F();
Tprime();
}
}
```

```
//
----------------------------------------------------------------------------------------------------
----------------
void F()
{
if(str[curr] == '(')
{
curr++;
E();
if(str[curr] == ')')
{
curr++;
return;
}
else
invalid();
}
else if(str[curr] == 'i')
{
curr++;
return;
}
else
invalid();
}
//
----------------------------------------------------------------------------------------------------
----------------
int main()
{
printf("Enter String: ");
scanf("%s", str);
E();
if(str[curr] == '$')
valid();
else
// printf("%c\n", str[curr]);
invalid();
}
```

Output :

P1)
S -> a | > | (T)
T -> T, S|S

program :

```
/*
S->a | > | ( T )
T->T, S|S
after removing lr
S->a | > | ( T )
T -> ST'
T' -> ,ST' | empty
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int curr = 0;
char str[100];
void S();
void T();
void Tprime();
void invalid()
{
printf("-----------------ERROR!----------------\n");
exit(0);
}
void valid()
{
```

```c
printf("----------------SUCCESS!---------------\n");
exit(0);
}
void S()
{
if (str[curr] == 'a' || str[curr] == '>')
{
curr++;
return;
}
else if (str[curr] == '(')
{
curr++;
T();
if (str[curr] == ')')
{
curr++;
return;
}
else
invalid();
}
else
invalid();
}
void T()
{
S();
Tprime();
}
void Tprime()
{
if (str[curr] == ',')
{
curr++;
S();
Tprime();
}
}
void main()
{
printf("Enter String: \n");
scanf("%s", str);
S();
if (str[curr] == '$')
```

```
valid();
else
invalid();
}
```

Output :

P2)
S -> UVW
U -> (S) | aSb | d
V -> aV |
W -> cW |

program :

```
/*
S->UVW
U -> (S) | aSb | d
V ->aV | empty
W->cW | empty
*/
#include <stdio.h>
#include <stdlib.h>
```

```c
#include <string.h>
int curr = 0;
char str[100];
void S();
void U();
void V();
void W();
void invalid()
{
printf("-----------------ERROR!----------------\n");
exit(0);
}
void valid()
{
printf("----------------SUCCESS!---------------\n");
exit(0);
}
void S()
{
U();
V();
W();
}
void U()
{
if (str[curr] == '(')
{
curr++;
S();
if (str[curr] == ')')
{
curr++;
return;
}
else
invalid();
}
else if (str[curr] == 'a')
{
curr++;
S();
if (str[curr] == 'b')
{
curr++;
return;
```

```c
}
else
invalid();
}
else if (str[curr] == 'd')
{
curr++;
return;
}
else
invalid();
}
void V()
{
if (str[curr] == 'a')
{
curr++;
V();
}
}
void W()
{
if (str[curr] == 'c')
{
curr++;
W();
}
}
void main()
{
printf("Enter String: \n");
scanf("%s", str);
S();
if (str[curr] == '$')
valid();
else
invalid();
}
```

Output :

P3)
S -> aAcBe
A -> Ab | b
B -> d

Program :

```
/*
S->aAcBe
A->Ab|b
B->d
after removing lr
S->aAcBe
A->bA'
A'->bA'|empty
B->d
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int curr = 0;
char str[100];
void S();
```

```c
void A();
void Aprime();
void B();
void invalid()
{
printf("-----------------ERROR!----------------\n");
exit(0);
}
void valid()
{
printf("----------------SUCCESS!---------------\n");
exit(0);
}
void S()
{
if (str[curr] == 'a')
{
curr++;
A();
if (str[curr] == 'c')
{
curr++;
B();
if (str[curr] == 'e')
{
curr++;
return;
}
else
invalid();
}
else
invalid();
}
else
invalid();
}
void A()
{
if (str[curr] == 'b')
{
curr++;
Aprime();
}
else
```

```c
invalid();
}
void Aprime()
{
if (str[curr] == 'b')
{
curr++;
Aprime();
}
}
void B()
{
if (str[curr] == 'd')
{
curr++;
return;
}
else
invalid();
}
void main()
{
printf("Enter String: \n");
scanf("%s", str);
S();
if (str[curr] == '$')
valid();
else
invalid();
}
```

Output :

P4)
S -> (L) | a
L -> L,S | S

program :

```
/*
S ->(L) | a
L ->L,S | S
after removing lr
S ->(L) | a
L ->SL'
L'->,SL'|empty
*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int curr = 0;
char str[100];
void S();
void L();
void Lprime();
void invalid()
{
printf("----------------ERROR!---------------\n");
exit(0);
}
void valid()
{
printf("---------------SUCCESS!--------------\n");
exit(0);
}
void S()
{
if (str[curr] == 'a')
{
curr++;
return;
}
else if (str[curr] == '(')
{
curr++;
L();
```

```c
if (str[curr] == ')')
{
        curr++;
return;
}
else
invalid();
}
else
invalid();
}
void L()
{
S();
Lprime();
}

void Lprime()
{
if (str[curr] == ',')
{
curr++;
S();
Lprime();
}
}

void main()
{
printf("Enter String: \n");
scanf("%s", str);
S();
if (str[curr] == '$')
valid();
else
invalid();
}
```

Output :

```
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/5th_sem_LABS/CD_LAB/
lab 6$ gcc -g lab6_p4.c -o p4
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/5th_sem_LABS/CD_LAB/
lab 6$ ./p4
Enter String:
(a,a)$
---------------SUCCESS!---------------
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/5th_sem_LABS/CD_LAB/
lab 6$ ./p4
Enter String:
(aa)$
----------------ERROR!----------------
rajvardhan@rajvardhan-HP-Pavilion-Laptop-15-cc1xx:~/Desktop/5th_sem_LABS/CD_LAB/
lab 6$
```