

COMPILER DESIGN LAB 6

Date @February 17, 2026

Name: Devadathan N R

Reg No: 230905010

Class: CSE - A1

Roll No: 04

Recursive Descent Parser

Question 1

$S \rightarrow a \mid > \mid (T)$

$T \rightarrow T, S \mid S$

Grammar without left recursion

$S \rightarrow a \mid > \mid (T)$

$T \rightarrow ST'$

$T' \rightarrow , ST' \mid \text{epsilon}$

CODE:

I6q1.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

int ptr=0;
char str[100];

void S();
```

```

void T();
void Tprime();

void reject(){
    printf("ERROR. String Rejected.\n");
    exit(0);
}

void accept(){
    printf("SUCCESS. String Accepted.\n");
    exit(0);
}

void S(){
    if(str[ptr] == 'a'){
        ptr++;
        return;
    }
    else if(str[ptr] == '>'){
        ptr++;
        return;
    }
    else if(str[ptr] == '('){
        ptr++;
        T();
        if(str[ptr] == ')'){
            ptr++;
            return;
        }
        else reject();
    }
    else reject();
}

void T(){
    S();
}

```

```

Tprime();
}

void Tprime(){
    if(str[ptr] == ','){
        ptr++;
        S();
        Tprime();
    }
    return;
}

int main(){
    printf("Enter string: ");
    scanf("%s", str);
    S();
    if(str[ptr] == '$') accept();
    else reject();
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ gcc l6q1.c -o l6q1
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q1
Enter string: ((a,>),a)$
SUCCESS. String Accepted.
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q1
Enter string: ((a,>),)$
ERROR. String Rejected.

```

Question 2

$S \rightarrow UVW$
 $U \rightarrow (S) \mid aSb \mid d$
 $V \rightarrow aV \mid \epsilon$
 $W \rightarrow cW \mid \epsilon$

No left recursion

CODE:

I6q2.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int ptr = 0;
char str[100];
void S();
void U();
void V();
void W();

void reject() {
    printf("ERROR. String Rejected.\n");
    exit(0);
}
void accept(){
    printf("SUCCESS. String Accepted.\n");
    exit(0);
}
void S(){
    U();
    V();
    W();
}
void U(){
    if(str[ptr] == '('){
        ptr++;
        S();
        if(str[ptr] == ')'){

```

```

        ptr++;
        return;
    }else reject();
}
else if(str[ptr] == 'a'){
    ptr++;
    S();
    if(str[ptr] == 'b'){
        ptr++;
        return;
    }else reject();
}
else if(str[ptr] == 'd') {
    ptr++;
    return;
}
else reject();
}
void V(){
    if(str[ptr] == 'a'){
        ptr++;
        V();
    }
    return;
}
void W(){
    if(str[ptr] == 'c'){
        ptr++;
        W();
    }
    return;
}
int main(){
    printf("Enter string: ");
    scanf("%s", str);
    S();
}

```

```

        if(str[ptr] == '$') accept();
        else reject();
    }

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ gcc l6q2.c -o l6q2
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q2
Enter string: (adb)aaaccc$
SUCCESS. String Accepted.
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q2
Enter string: (adb)acaa$
ERROR. String Rejected.

```

Question 3

$S \rightarrow aAcBe$

$A \rightarrow Ab|b$

$B \rightarrow d$

Grammar without left recursion

$S \rightarrow aAcBe$

$A \rightarrow bA'$

$A' \rightarrow bA' | \epsilon$

$B \rightarrow d$

CODE:

l6q3.c

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int ptr = 0;
char str[100];

```

```

void S();
void A();
void Aprime();
void B();

void reject() {
    printf("ERROR. String Rejected.\n");
    exit(0);
}
void accept(){
    printf("SUCCESS. String Accepted.\n");
    exit(0);
}

void S(){
    if(str[ptr] == 'a'){
        ptr++;
        A();
        if(str[ptr] == 'c'){
            ptr++;
            B();
            if(str[ptr] == 'e'){
                ptr++;
                return;
            }else reject();
        }else reject();
    }else reject();
}

void A(){
    if(str[ptr] == 'b'){
        ptr++;
        Aprime();
    }else reject();
}

```

```

void Aprime(){
    if(str[ptr] == 'b'){
        ptr++;
        Aprime();
    }
    return;
}

void B(){
    if(str[ptr] == 'd'){
        ptr++;
        return;
    }
}

int main(){
    printf("Enter string: ");
    scanf("%s", str);
    S();
    if(str[ptr] == '$') accept();
    else reject();
}

```

OUTPUT:

```

CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ gcc l6q3.c -o l6q3
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q3
Enter string: abbbcd$e
SUCCESS. String Accepted.
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q3
Enter string: abbbcd$e
ERROR. String Rejected.

```

Question 4

$$S \rightarrow (L) \mid a$$

$$L \rightarrow L, S \mid S$$

Grammar without left recursion

$S \rightarrow (L) \mid a$
 $L \rightarrow SL'$
 $L' \rightarrow ,SL' \mid \epsilon$

CODE:

I6q4.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int ptr = 0;
char str[100];

void S();
void L();
void Lprime();

void reject() {
    printf("ERROR. String Rejected.\n");
    exit(0);
}
void accept(){
    printf("SUCCESS. String Accepted.\n");
    exit(0);
}

void S(){
    if(str[ptr] == '('){
        ptr++;
        L();
        if(str[ptr] == ')'){
            ptr++;
        }
    }
}
```

```

        return;
    }else reject();
}
else if(str[ptr] == 'a'){
    ptr++;
    return;
}
else reject();
}

void L(){
    S();
    Lprime();
}

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

int main(){
    printf("Enter string: ");
    scanf("%s", str);
    S();
    if(str[ptr] == '$') accept();
    else reject();
}

```

OUTPUT:

```
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ gcc l6q4.c -o l6q4
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q4
Enter string: ((a,a),a)$
SUCCESS. String Accepted.
CD_A1@CL3-02:~/Desktop/230905010_DevadathanNR_CS_A1/06_LAB$ ./l6q4
Enter string: ((a,a),)$
ERROR. String Rejected.
```
