AASHNA NITIN KUNKOLIENKER
190905304
CSE-D-44
CD lab 6

Question 1)

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

Before parsing, we need to remove left recursion.

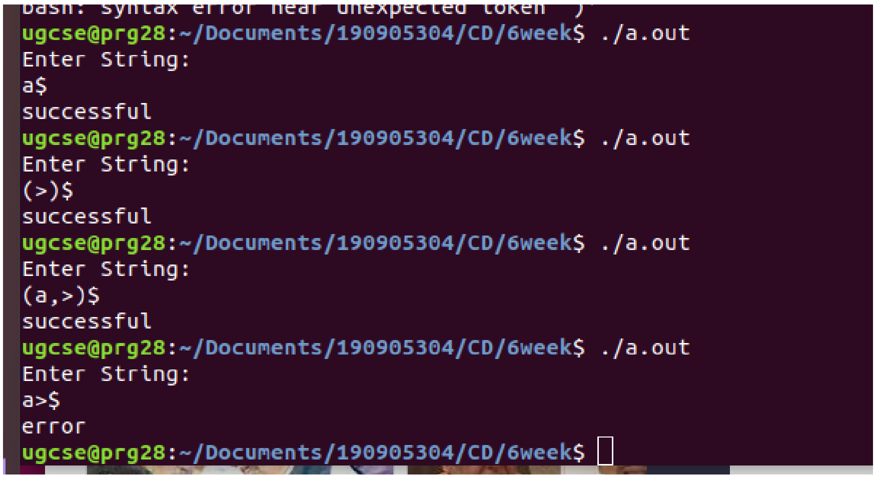
New grammar:

S->a | > | (T)
T -> ST'
T' -> ,ST' | **ε**

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

void S();
void T();
void Tprime();

int current = 0;
char str[100];


void invalid()
{
   printf("error\n");
   exit(0);
}
void valid()
{
   printf("successful\n");
   exit(0);
}
void S()
{
   //printf("here %s",str);
   if (str[current] == 'a' || str[current] == '>')
   {
      current++;
```

```
        return;
    }
    else if (str[current] == '(')
    {
        current++;
        T();
        if (str[current] == ')')
        {
            current++;
            return;
        }
        else
        {
            invalid();
            printf("1");
        }
    }
    else
    {
        invalid();
                    printf("2");

    }
}
void T()
{
    S();
    Tprime();
}
void Tprime()
{
    if (str[current] == ',')
    {
        current++;
        S();
        Tprime();
    }
}
void main()
{
    printf("Enter String: \n");
    scanf("%s", str);
    S();
    if (str[current] == '$')
    {
        valid();
    }
    else
    {
        invalid();
    }
}
```

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

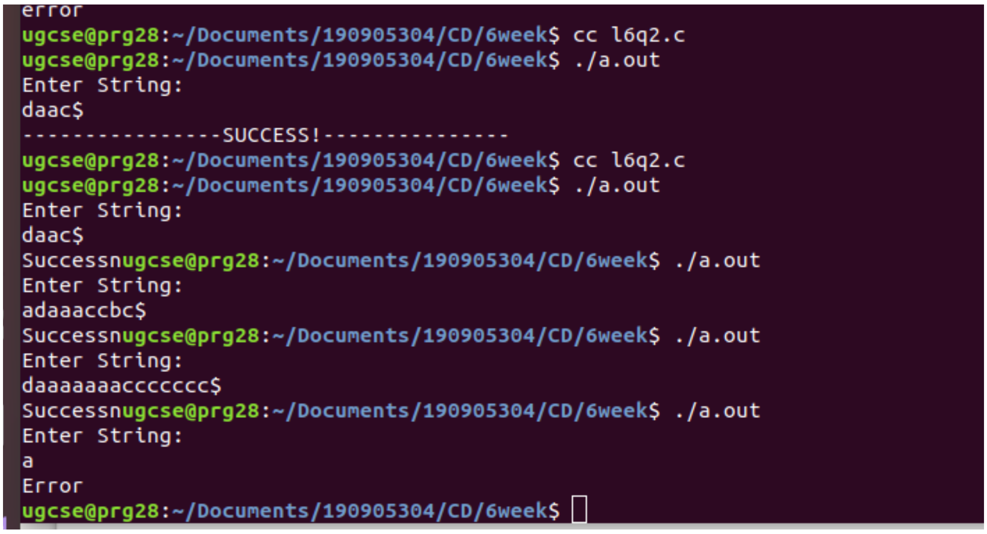


```
bash: syntax error near unexpected token `)'
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
a$
successful
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
(>)$
successful
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
(a,>)$
successful
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
a>$
error
ugcse@prg28:~/Documents/190905304/CD/6week$
```

Question 2)

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

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int current = 0;
char str[100];
void S();
void U();
void V();
void W();
void invalid()
{
   printf("Error\n");
   exit(0);
}
void valid()
{
   printf("Successn");
   exit(0);
}
void S()
{
   U();Question 2)
   V();
   W();
}
void U()
{
   if (str[current] == '(')
   {
      current++;
      S();
      if (str[current] == ')')
      {
         current++;
         return;
      }
      else
         invalid();
   }
   else if (str[current] == 'a')
   {
      current++;
      S();
      if (str[current] == 'b')
      {
            current++;
            return;
        }
        else
            invalid();
    }
    else if (str[current] == 'd')
    {
        current++;
        return;
    }
    else
        invalid();
}
void V()
{
    if (str[current] == 'a')
    {
        current++;
        V();
    }
}
void W()
{
    if (str[current] == 'c')
    {
        current++;
        W();
    }
}
void main()
{
    printf("Enter String: \n");
    scanf("%s", str);
    S();
    if (str[current] == '$')
        valid();
    else
        invalid();
}
```



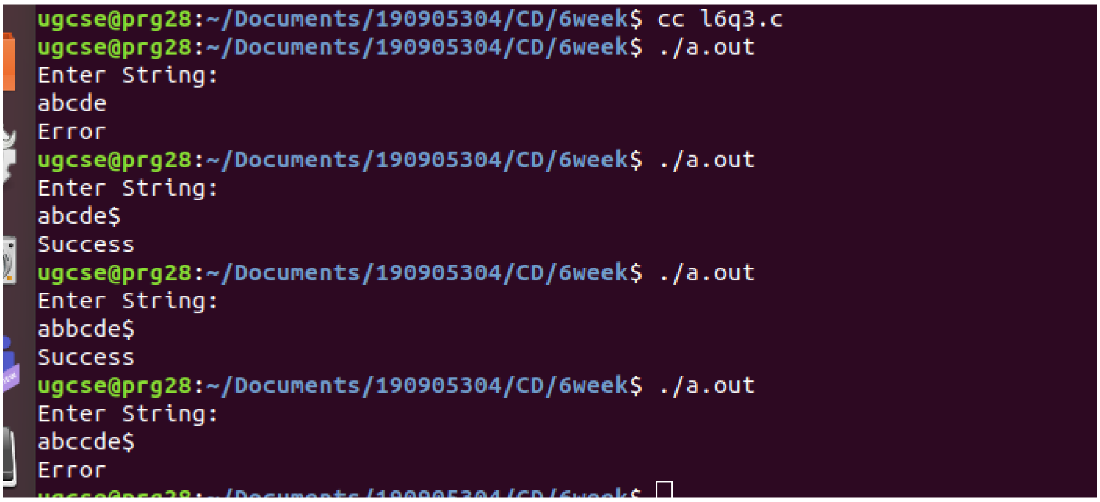
```
error
ugcse@prg28:~/Documents/190905304/CD/6week$ cc l6q2.c
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
daac$
----------------SUCCESS!---------------
ugcse@prg28:~/Documents/190905304/CD/6week$ cc l6q2.c
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
daac$
Successnugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
adaaaccbc$
Successnugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
daaaaaaaccccccc$
Successnugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
a
Error
ugcse@prg28:~/Documents/190905304/CD/6week$ 
```

Question 3)

S->aAcBe
A->Ab|b
B->d
Before parsing, we need to remove left recursion.

New grammar:
S->aAcBe
A->bA'
A'->bA'|empty
B->d

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int current = 0;
char str[100];
void S();
void A();
void A2();
void B();
void invalid()
{
    printf("Error\n");
    exit(0);
}
void valid()
{
    printf("Error\n");
    exit(0);
}
void S()
{
    if (str[current] == 'a')
    {
        current++;
        A();
        if (str[current] == 'c')
        {
            current++;
            B();
            if (str[current] == 'e')
            {
                current++;
                return;
            }
            else
                invalid();
        }
        else
            invalid();
    }
    else
        invalid();
}
void A()
{
    if (str[curr] == 'b')
    {
        curr++;
        A2();
    else
        invalid();
}
void A2()
{
    if (str[current] == 'b')
    {
        current++;
        A2();
    }
}
void B()
{
    if (str[current] == 'd')
    {
        current++;
        return;
    }
    else
        invalid();
}
void main()
{
    printf("Enter String: \n");
    scanf("%s", str);
    S();
    if (str[current] == '$')
        valid();
    else
        invalid();
}
```



```
ugcse@prg28:~/Documents/190905304/CD/6week$ cc l6q3.c
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
abcde
Error
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
abcde$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
abbcde$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
abccde$
Error
ugcse@prg28:~/Documents/190905304/CD/6week$ 
```

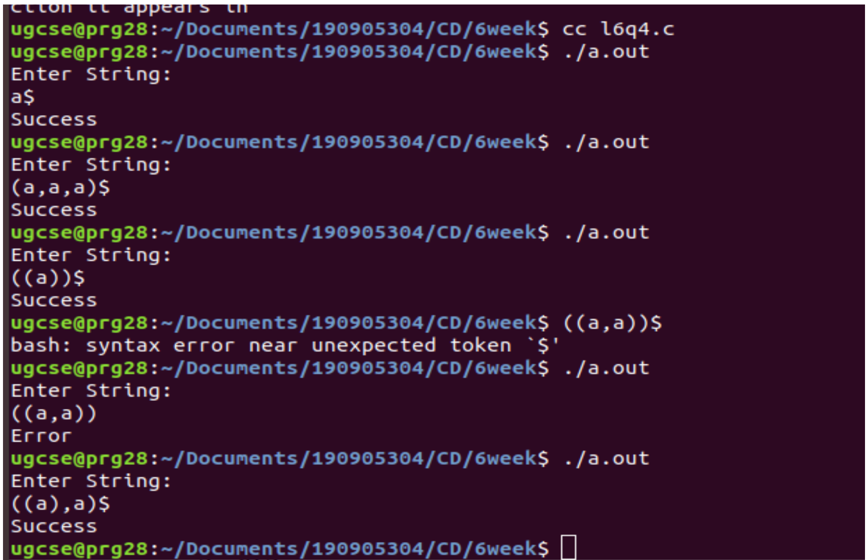
Question 4)

S ->(L) | a
L ->L,S | S
Before parsing, we need to remove left recursion.

New grammar:
S ->(L) | a
L ->SL'
L'->,SL'| **ε**

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

char str[200];
int current = 0;



void S();
void L();
void L2();
void invalid()
{
   printf("Error\n");
   exit(0);
}
void valid()
{
   printf("Success\n");
   exit(0);
}
void S()
{
   if (str[current] == 'a')
```

```
    {
        current++;
        return;
    }
    else if (str[current] == '(')
    {
        current++;
        L();
        if (str[current] == ')')
        {
            current++;
            return;
        }
        else
            invalid();
    }
    else
        invalid();
}
void L()
{
    S();
    L2();
}
void L2()
{
    if (str[current] == ',')
    {
        current++;
        S();
        L2();
    }
}
void main()
{
    printf("Enter String: \n");
    scanf("%s", str);
    S();
    if (str[current] == '$')
        valid();
    else
        invalid();
}
```



```
ction it appears in
ugcse@prg28:~/Documents/190905304/CD/6week$ cc l6q4.c
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
a$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
(a,a,a)$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
((a))$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ ((a,a))$
bash: syntax error near unexpected token `$'
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
((a,a))
Error
ugcse@prg28:~/Documents/190905304/CD/6week$ ./a.out
Enter String:
((a),a)$
Success
ugcse@prg28:~/Documents/190905304/CD/6week$ 
```