SRM UNIVERSITY AP
Andhra Pradesh

Department of Computer Science and Engineering
Compiler Design Lab (CS 306)

Week 7: Implementation of LL(1) parser using C

Week 7 Program

1. Implement non-recursive Predictive Parser for the grammar

   S -> aBa
   B -> bB | ε

	a	b	$
S	S→aBa		
B	B→ε	B→bB	

2. **Lab Assignment:** Implement Predictive Parser using C for the Expression Grammar

   E → TE’
   E’→ +TE’ | ε
   T → FT’
   T’→ *FT’ | ε
   F → (E) | d

Instructions:

- Explanation and code of first program explaining the requirements in the program are given below.
- You are required to implement second one on your own and upload both into your Github accounts under the folder **Week7-Lab-exercise**

Programs:

Code of first program:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
int i=0,top=0;
char stack[20],ip[20];

void push(char c)
{
    if (top>=20)
            printf("Stack Overflow");
    else
            stack[top++]=c;
}
```

```
void pop(void)
{
    if(top<0)
            printf("Stack underflow");
    else
            top--;
}

void error(void)
{
printf("\n\nSyntax Error!!!! String is invalid\n");
exit(0);
}

int main()
{
int n;

printf("The given grammar is\n\n");
printf("S -> aBa\n");
printf("B -> bB | epsilon \n\n");
printf("Enter the string to be parsed:\n");
scanf("%s",ip);
n=strlen(ip);
ip[n]='$';
ip[n+1]='\0';
push('$');
push('S');
while(ip[i]!='\0')
{ if(ip[i]=='$' && stack[top-1]=='$')
  {
    printf("\n\n Successful parsing of string \n");
    return 1;
  }
  else
    if(ip[i]==stack[top-1])
    {
      printf("\nmatch of %c ",ip[i]);
      i++;pop();
    }
    else
    {
            if(stack[top-1]=='S' && ip[i]=='a')
            {
                printf(" \n S ->aBa");
                pop();
                push('a');
                push('B');
                push('a');
            }
            else
            if(stack[top-1]=='B' && ip[i]=='b')
```

```
                        {
                                printf("\n B ->bB");
                                pop();push('B');push('b');
                        }
                            else
                              if(stack[top-1]=='B' && ip[i]=='a')
                              {
                                  printf("\n B -> epsilon");
                                  pop();
                              }
                              else
                                  error();
               }
        }
       }//end of main

       }
```

Testcases: Test your program with test cases covering all requirements.