Compiler Design  
Lab Exercise 7

short line

**Aim: Perform Shift Reduce Parsing on a Given String**

**Algorithm:**

1) Shift reduce parsing performs the two actions: shift and reduce. ...

2) At the shift action, the current symbol in the input string is pushed to a stack.

3) At each reduction, the symbols will be replaced by the non-terminals.

**Code:(Language : Python3)**

gram = {

"S":["aSb","cSd","e"]

}

starting\_terminal = "S"

inp = "aaebb$"

stack = "$"

print(f'{"Stack": <15}'+"|"+f'{"Input Buffer": <15}'+"|"+f'Parsing Action')

print(f'{"-":-<50}')

while True:

    action = True

    i = 0

    while i<len(gram[starting\_terminal]):

        if gram[starting\_terminal][i] in stack:

            stack = stack.replace(gram[starting\_terminal][i],starting\_terminal)

            print(f'{stack: <15}'+"|"+f'{inp: <15}'+"|"+f'Reduce S->{gram[starting\_terminal][i]}')

            i=-1

            action = False

        i+=1

    if len(inp)>1:

            stack+=inp[0]

            inp=inp[1:]

            print(f'{stack: <15}'+"|"+f'{inp: <15}'+"|"+f'Shift')

            action = False

    if inp == "$" and stack == ("$"+starting\_terminal):

            print(f'{stack: <15}'+"|"+f'{inp: <15}'+"|"+f'Accepted')

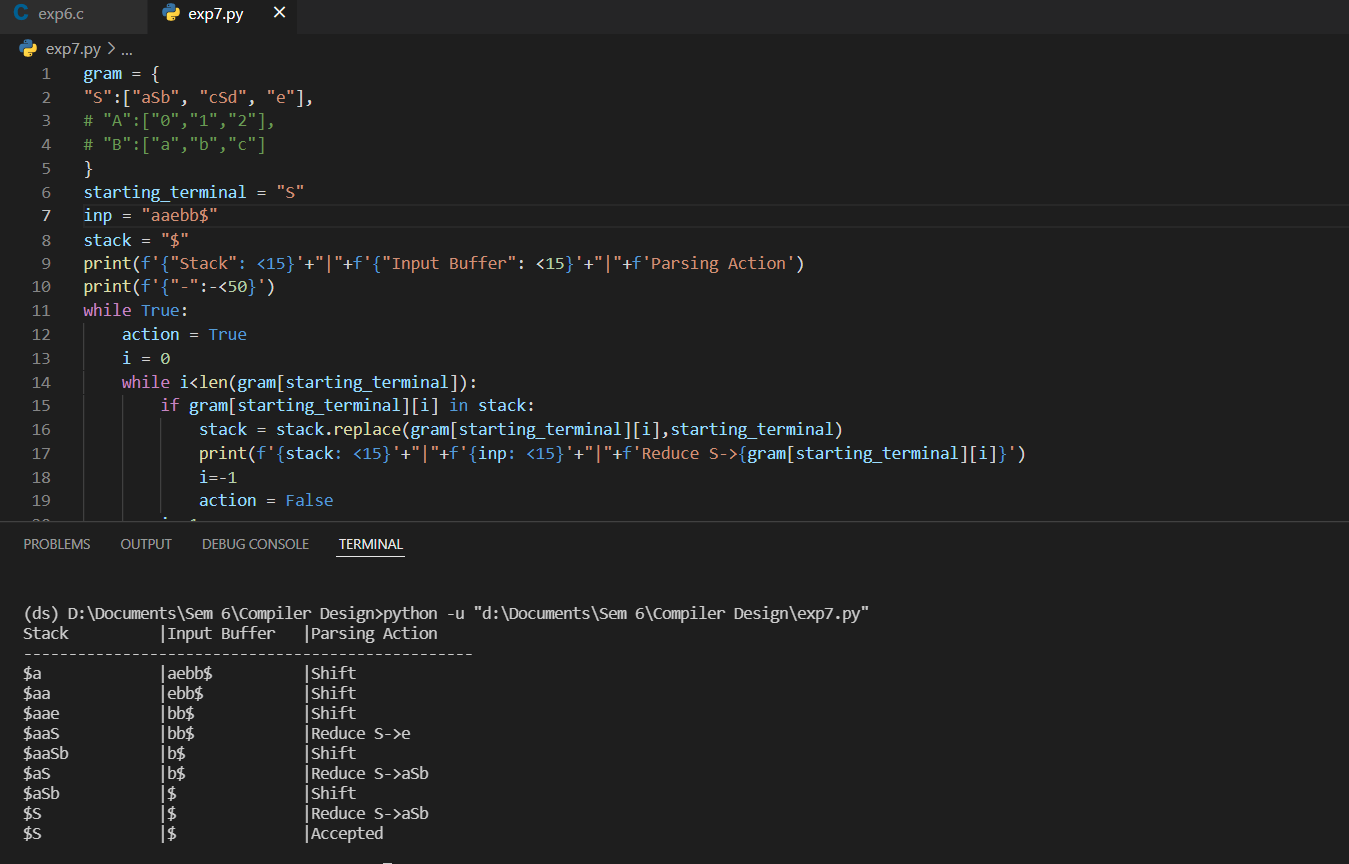
            break

    if action:

            print(f'{stack: <15}'+"|"+f'{inp: <15}'+"|"+f'Rejected')

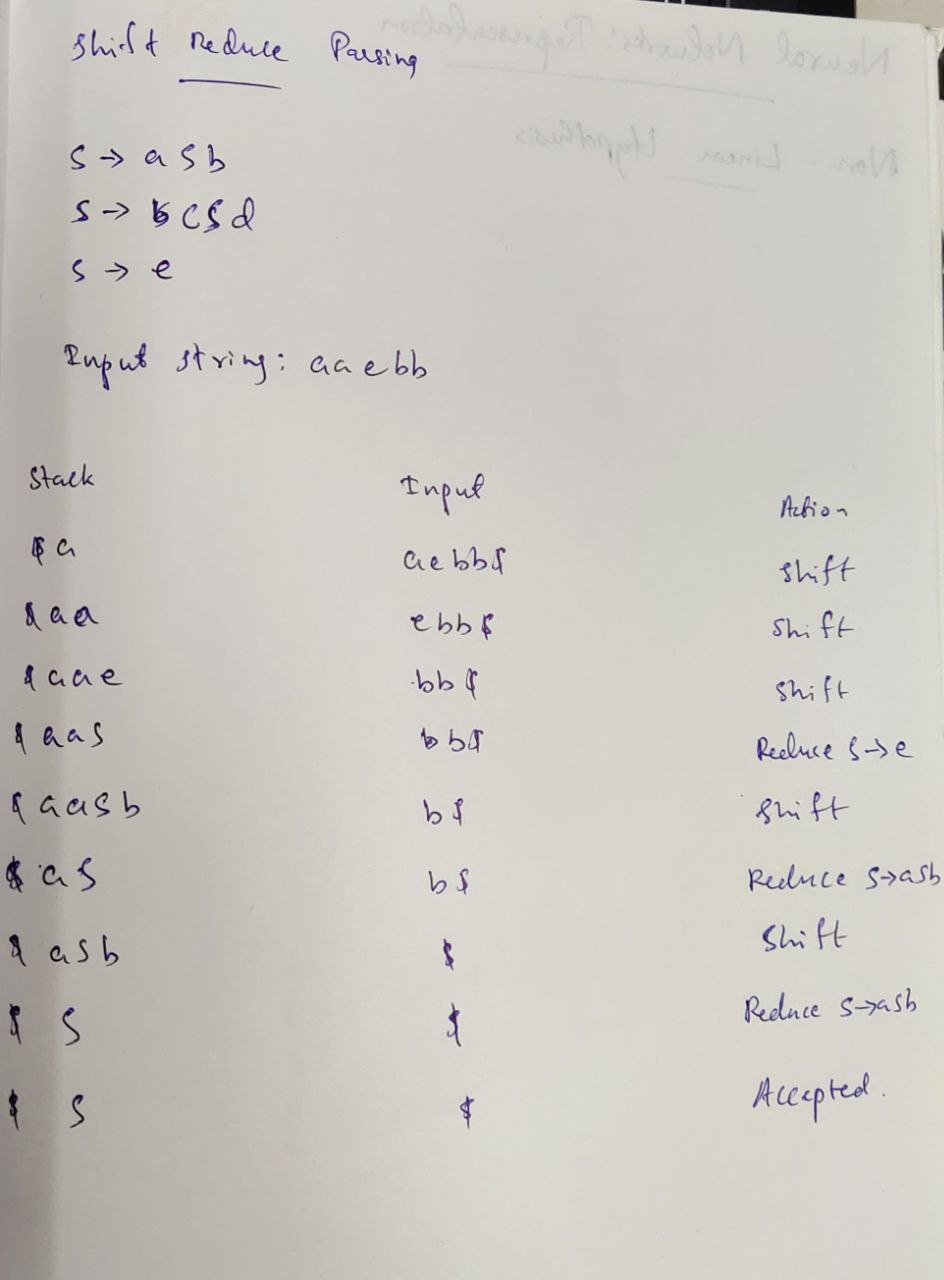
            break

**Output:**



**Manual Output:**

We compare the output from code as well as handwritten and deduce that they are the same for the given production rules.



**Result:** Implementation of Shift Reduce Parsing for the given grammar has been completed and verified by Manual Output.