**COMPILER DESIGN PRACTICAL LAB-7**

**Shift Reduce Parsing**

SANDRA MARIA TONY RA1911026010045 K1

**AIM:** A program for construction of **Shift Reduce Parsing Table**

**THEORY:**

**Shift Reduce parser** attempts for the construction of parse in a similar manner as done in bottom-up parsing i.e. the parse tree is constructed from leaves(bottom) to the root(up). A more general form of the shift-reduce parser is the LR parser. 

**ALGORITHM:**

1. Start

2. Ask the user to enter the set of productions.

3. Construct the parsing table.

4. Display the output.

5. Exit

**CODE:**

#include <iostream>

#include<string.h>

using namespace std;

struct grammer{

char p[20];

char prod[20];

}g[10];

int main()

{

int i,stpos,j,k,l,m,o,p,f,r;

int np,tspos,cr;

cout<<"\nEnter Number of productions:";

cin>>np;

char sc,ts[10];

cout<<"\nEnter productions:\n";

for(i=0;i<np;i++)

{

cin>>ts;

strncpy(g[i].p,ts,1);

strcpy(g[i].prod,&ts[3]);

}

char ip[10];

cout<<"\nEnter Input:";

cin>>ip;

int lip=strlen(ip);

char stack[10];

stpos=0;

i=0;

sc=ip[i];

stack[stpos]=sc;

i++;stpos++;

cout<<"\n\nStack\tInput\tAction";

do

{

r=1;

while(r!=0)

{

cout<<"\n";

for(p=0;p<stpos;p++)

{

cout<<stack[p];

}

cout<<"\t";

for(p=i;p<lip;p++)

{

cout<<ip[p];

}

if(r==2)

{

cout<<"\tReduced";

}

else

{

cout<<"\tShifted";

}

r=0;

for(k=0;k<stpos;k++)

{

f=0;

for(l=0;l<10;l++)

{

ts[l]='\0';

}

tspos=0;

for(l=k;l<stpos;l++)

{

ts[tspos]=stack[l];

tspos++;

}

for(m=0;m<np;m++)

{

cr = strcmp(ts,g[m].prod);

if(cr==0)

{

for(l=k;l<10;l++)

{

stack[l]='\0';

stpos--;

}

stpos=k;

strcat(stack,g[m].p);

stpos++;

r=2;

}

}

}

}

sc=ip[i];

stack[stpos]=sc;

i++;stpos++;

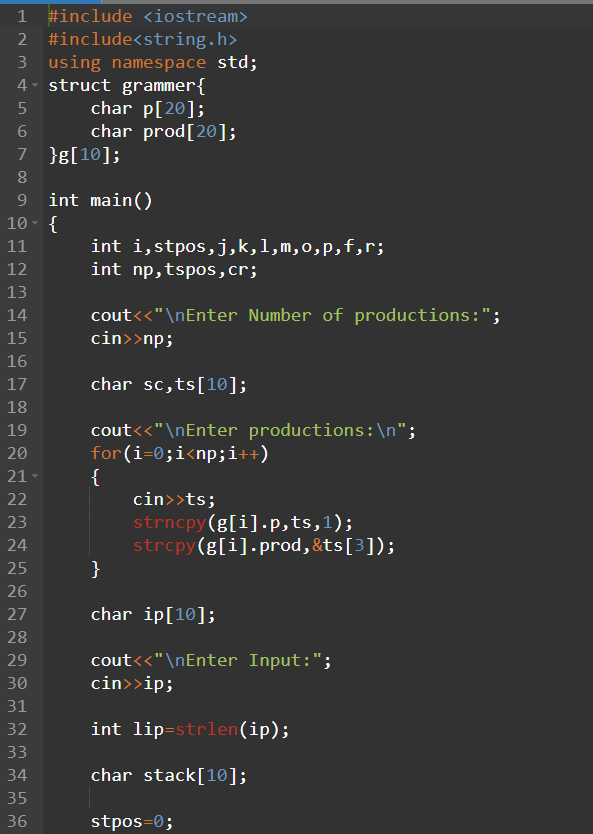
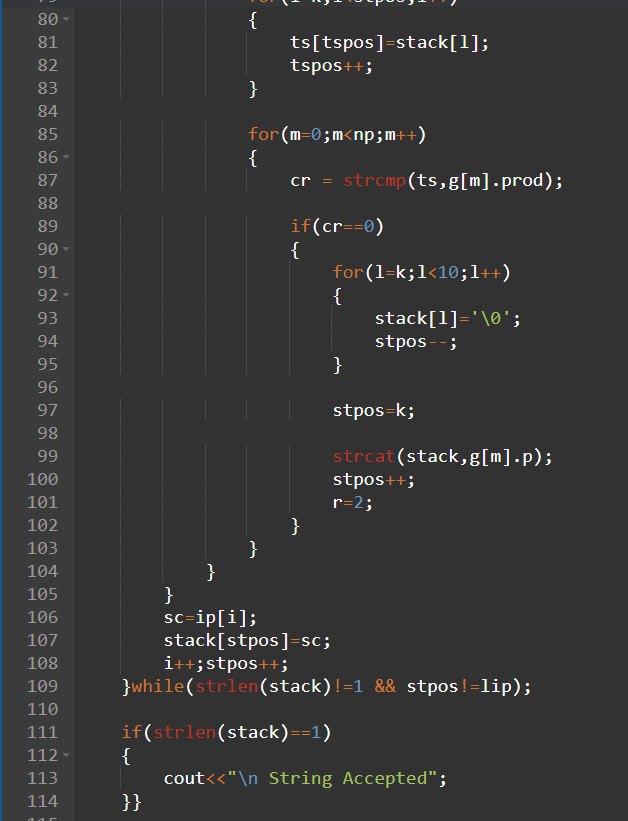
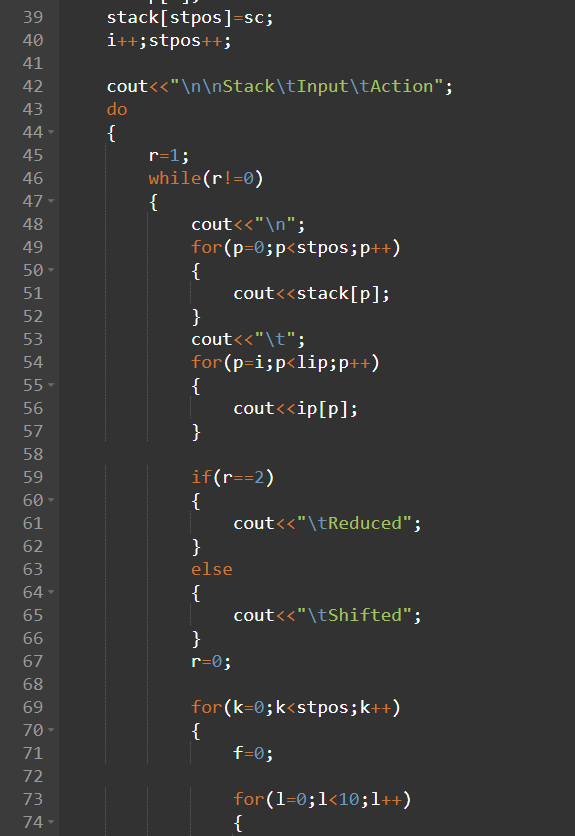
}while(strlen(stack)!=1 && stpos!=lip);

if(strlen(stack)==1)

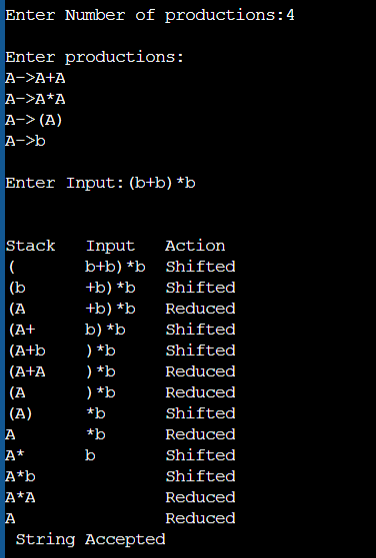
{

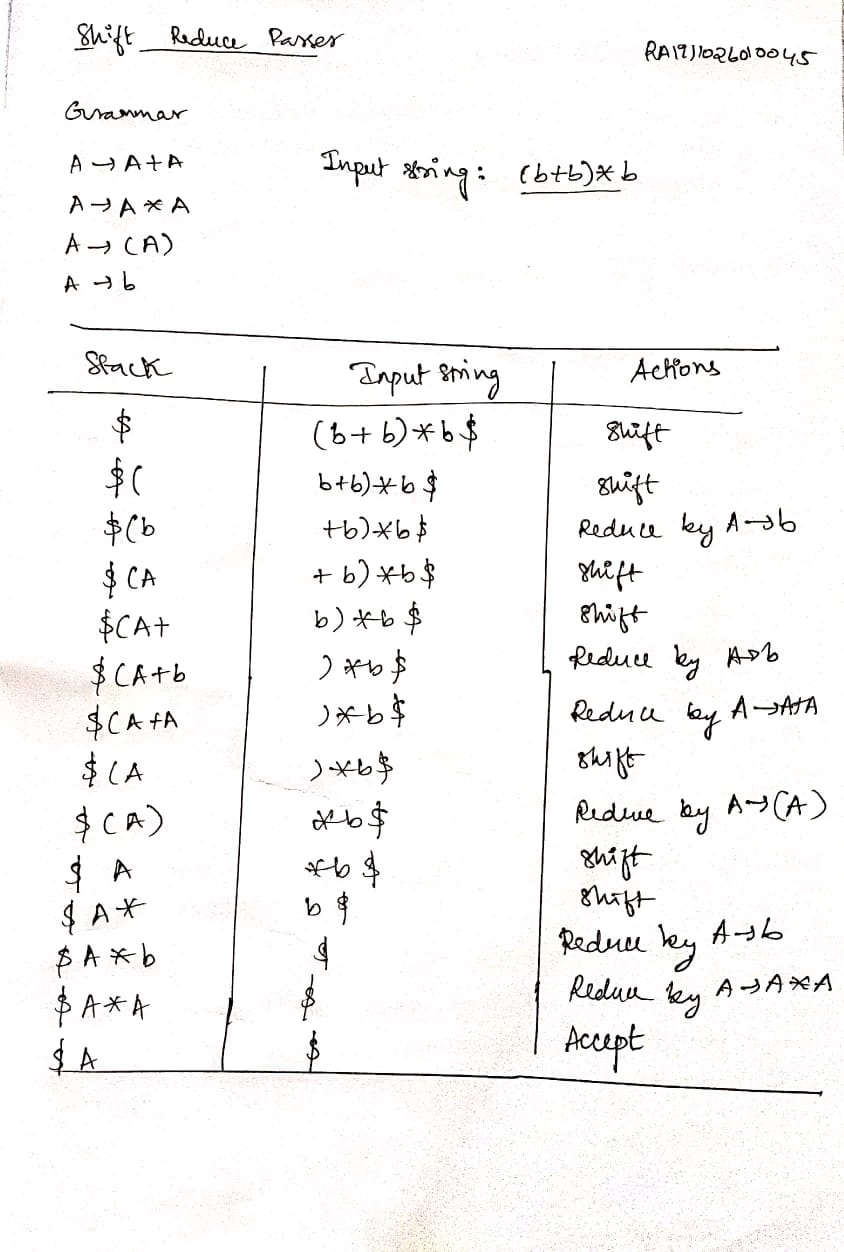
cout<<"\n String Accepted";

}}

**Output:**



**MANUAL CALCULATION:** 

kjhkjh

**RESULT:** The code for construction of **Shift Reduce Parsing** produced same output as the result from manual calculation thus the program was executed successfully.