**CD LAB ASSIGNMENT WEEK-10**

***Submitted by:***

***M. Lakshmi Abhigna***

***CSE-F***

***AP21110010343***

***Problem:***

Shift reduce parser for the grammar :

S-->AB

A-->aAa

A-->bAb

A-->c

B-->0B1

B-->01

***Code:***

#include<stdio.h>

#include<stdlib.h>

void pop(),push(char ),display();

char stack[100]="\0";

char inputbuffer[100];

int top=-1;

char \*ip;

int main()

{

printf("S--> AB\n");

printf("A-->aAa\n");

printf("A-->bAb\n");

printf("A-->c\n");

printf("B-->0B1\n");

printf("B-->01\n");

printf(" enter the input string followed by $ \n");

scanf("%s",inputbuffer);

ip=inputbuffer;

push('$');

printf("STACK\t BUFFER \t ACTION\n");

printf("-----\t ------- \t ------\n");

display();

do

{

if((stack[top]=='S' && stack[top-1]=='$') && (\*(ip)=='$'))

break;

if(stack[top]=='$')

{

push(\*ip);

ip++;

printf("\tShift");

}

else if(stack[top]=='B' && stack[top-1]=='A')

{

display();

pop();

pop();

push('S');

printf("\tReduce S->AB\n");

}

else if(stack[top]=='c' && \*ip!='b' && \*ip!='a')

{

display();

pop();

push('A');

printf("\tReduce A->c\n");

}

else if(stack[top]=='1' && stack[top-1]=='0' && stack[top-2]!='0' && stack[top-2]!=1 )

{

display();

pop();

pop();

push('B');

printf("\tReduce B->01\n");

}

else if(stack[top]=='1' && stack[top-1]=='1' && stack[top-2]=='0' && stack[top-3]=='0')

{

display();

pop();

pop();

pop();

pop();

push('B');

printf("\tReduce B->0B1\n");

}

else if(stack[top]=='a' && stack[top-1]=='c' && stack[top-2]=='a')

{

display();

pop();

pop();

pop();

push('A');

printf("Reduce A->aAa\n");

}

else if(stack[top]=='b' && stack[top-1]=='c' && stack[top-2]=='b')

{

display();

pop();

pop();

pop();

push('A');

printf("\tReduce A->bAb\n");

}

else if(stack[top]=='1' && stack[top-1]=='B' && stack[top-2]=='0')

{

display();

pop();

pop();

pop();

push('B');

printf("\tReduce B->0B1\n");

}

else if(stack[top]=='a' && stack[top-1]=='A' && stack[top-2]=='a')

{

display();

pop();

pop();

pop();

push('A');

printf("\tReduce A->aAa\n");

}

else if(stack[top]=='b' && stack[top-1]=='A' && stack[top-2]=='b')

{

display();

pop();

pop();

pop();

push('A');

printf("\tReduce A->bAb\n");

}

else

{

display();

push(\*ip);

if(\*ip=='\0'||\*ip=='\n')

{

printf("\tError\n");

getchar();

return 0;

}

ip++;

printf("\tshift");

}

}while(1);

display();

printf("\tAccept\n\n\n");

}

void push(char c)

{

top++;

stack[top]=c;

}

void pop()

{

stack[top]='\0';

top--;

}

void display()

{

printf("\n%s\t%s\t",stack,ip);

}

***Output:***

