LL PARSING

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
char table[10][10]={"NT","a","b","A","aBa","error","B","@","bB"};
char buffer[10],stack[10];
int top=-1;
char pop()
{
return stack[top--];
}
void push(int e) {
stack[++top]=e;
}
void display_stack() {
int i=top;
while(i>=0) {
printf("%c",stack[i]);
i--;
}
printf("\n");
}
char*parse_table(char stack_top,char input_val) {
switch(stack_top) {
case 'A' :
        switch(input_val) {
        case 'a' :
                return table[4];
        case 'b' :
                return table[5];
```

```
                }
                break;
        case 'B' :
        switch(input_val) {
                case 'a' :
                return table[7];
        case 'b' :
                return table[8];
                }
        default : return table[5];
        }
        }

int main()
 {
int ptr=0,i=0,j,k,w=0;
char*str;
                for(j=0;j<3;j++)
                {
                        for(k=0;k<3;k++)
                        {
                        printf("%s\t",table[w++]);
                        }
                printf("\n");
                }
        printf("enter the string\n");
        scanf("%s",buffer);
                if(buffer[strlen(buffer)-1]!=';')
                        {
                                printf("string should end with :");
                                exit(0);
```

```
                }
        push('$');
        push('A');
                while(stack[top]!='$' && (ptr<strlen(buffer))) {
                        if(stack[top]==buffer[ptr])
                        {
                                ptr++;
                                printf("1. element is popped  is
%c\n",pop());
                                }
                        else if(stack[top]=='@') {
                                printf(" element is popped  is %c\n",pop());
                                }
                        else {
                                str=parse_table(stack[top],buffer[ptr]);
                if(strcmp(str,"error")==0) {
                printf("error in
                parsing\n");break;
                }
                        printf("3. element is popped is
                        %c\n",pop());for(i=strlen(str)-1;i>=0;i--)
                        push(*(str+i));
                        }
                display_stack();
                }
        if(stack[top]=='$' &&
                buffer[ptr]==';')
                printf("string is
                accepted\n");
                        else {
                                printf("string is not accepted\n");
```

```
}
return 0;}
```