

/* B1: Write a program to implement

(a) Recursive Descent Parsing with back tracking (Brute Force Method)

$$S \rightarrow cAd$$
$$A \rightarrow ab / a$$

(b) Recursive Descent Parsing with back tracking (Brute Force Method)

$$S \rightarrow cAd$$
$$A \rightarrow a / ab$$

Note: What is the difference that you have experienced w.r.t. the language accepted for these two CFGs.
*/

/* s->cAd A->ab/a */

File: B1_a.cpp

```
/*      s->cAd      A->ab/a      */
#include<stdio.h>
#include<conio.h>
#include<string.h>
int A();
char str[15];
int isave, curr_ptr=0;
int main(void)
{
    //clrscr();
    printf("1.S->cAd\n2.A->ab/a\n");
    printf("this is parser for the above grammar:\n");
    printf("Enter any string:");
    scanf("%s", &str);
    while(curr_ptr<strlen(str))
    {
        //S has only one immediate derivation which is cAd

        //match with c
        if (str[curr_ptr]=='c')
        {
            curr_ptr++;
            //call function to match A
            if (A()) //checking the productions of A->ab/a
            {
                curr_ptr++;
                //match d
                if (str[curr_ptr]=='d' && str[curr_ptr+1]=='\0')
                {
                    //success
                    printf("string is accepted by the grammar");
                    getch();
                    return 1;
                }
            }
            else break;
        }
    }
}
```

```

        }
        else break;
    }
    else break;
}

    //incase any of them fail to match return negatively.
    printf("string is not accepted by the grammar");
    //getch();
    return 0;
}

int A()      //sub function A()
{
    //this function matches all terminal strings generated by the variable
    A. Here the only terminal strings generated by A is ab or a.
    isave=curr_ptr;

    //match with a and advance and match with b. If successful return
    positive.
    if (str[curr_ptr]=='a')
    {
        curr_ptr++;
        if(str[curr_ptr]=='b')
            return 1;
    }
    curr_ptr=isave; //return to start

    //check if a is matched and return accordingly.
    if(str[curr_ptr]=='a')
        return 1;
    else
        return 0;
}

```

```
1 // g++ g1.cpp -std=c++11 -g
2 #include <iostream>
3 #include <string>
4 #include <string.h>
5 #include <conio.h>
6 #include <stdio.h>
7 #include <stdlib.h>
8 #include <ctype.h>
9 #include <math.h>
10 #include <algorithm>
11 #include <vector>
12 #include <map>
13 #include <set>
14 #include <stack>
15 #include <queue>
16 #include <deque>
17 #include <list>
18 #include <iterator>
19 #include <functional>
20 #include <numeric>
21 #include <memory>
22 #include <limits>
23 #include <climits>
24 #include <cerr>
25 #include <csignal>
26 #include <ctime>
27 #include <cstdlib>
28 #include <cstring>
29 #include <cstdio>
30 #include <cmath>
31 #include <cassert>
32 #include <cassert>
33 #include <cassert>
34 #include <cassert>
35 #include <cassert>
36 #include <cassert>
37 #include <cassert>
38 #include <cassert>
39 #include <cassert>
40 #include <cassert>
41 #include <cassert>
42 #include <cassert>
43 #include <cassert>
44 #include <cassert>
45 #include <cassert>
46 #include <cassert>
47 #include <cassert>
48 #include <cassert>
49 #include <cassert>
50 #include <cassert>
51 #include <cassert>
52 #include <cassert>
53 #include <cassert>
54 #include <cassert>
```

Compiler (1) Resources Compile Log Debug Find Results Close

Line Col File Message

D:\NTW_CD_Lab\CompilerDesignPrograms\Set_8_Programs\B1_a.cpp [Warning] command line option '-std=c99' is valid for C/C++, but not for C++

Line: 41 Col: 3 Sel: 0 Lines: 54 Length: 1018 Insert Done parsing in 0.172 seconds

```
1 // g++ g1.cpp -std=c++11 -g
2 #include <iostream>
3 #include <string>
4 #include <string.h>
5 #include <conio.h>
6 #include <stdio.h>
7 #include <stdlib.h>
8 #include <ctype.h>
9 #include <math.h>
10 #include <algorithm>
11 #include <vector>
12 #include <map>
13 #include <set>
14 #include <stack>
15 #include <queue>
16 #include <deque>
17 #include <list>
18 #include <iterator>
19 #include <functional>
20 #include <numeric>
21 #include <memory>
22 #include <limits>
23 #include <climits>
24 #include <cerr>
25 #include <csignal>
26 #include <ctime>
27 #include <cstdlib>
28 #include <cstring>
29 #include <cstdio>
30 #include <cmath>
31 #include <cassert>
32 #include <cassert>
33 #include <cassert>
34 #include <cassert>
35 #include <cassert>
36 #include <cassert>
37 #include <cassert>
38 #include <cassert>
39 #include <cassert>
40 #include <cassert>
41 #include <cassert>
42 #include <cassert>
43 #include <cassert>
44 #include <cassert>
45 #include <cassert>
46 #include <cassert>
47 #include <cassert>
48 #include <cassert>
49 #include <cassert>
50 #include <cassert>
51 #include <cassert>
52 #include <cassert>
53 #include <cassert>
54 #include <cassert>
```

Compiler (1) Resources Compile Log Debug Find Results Close

Line Col File Message

D:\NTW_CD_Lab\CompilerDesignPrograms\Set_8_Programs\B1_a.cpp [Warning] command line option '-std=c99' is valid for C/C++, but not for C++

Line: 31 Col: 9 Sel: 0 Lines: 54 Length: 1018 Insert Done parsing in 0.172 seconds

File: B1_b.cpp

```
/*
    The Production are:
                                S -> cAd
    A -> a/ab
*/

#include <stdio.h>
#include <conio.h>
#include <ctype.h>
int S ();
int A ();
int i=0;
char S1[10];
int main()
{
    int j=0;
    //clrscr();
    printf(" Enter Input String \n ");
    gets(S1);
    j=S ();
    if(j) {
        printf( " String is Accepted \n ");
    }
    else {
        printf(" String is Rejected \n" );
    }
    getch();
}

int S ()
{
    //S has only one immediate derivation which is cAd

    //match with c
    if(S1[i] == 'c')
    {
        i++;
        //call function to match A
        if (A ())
        {
            //match d
            if(S1[i] == 'd')
            {
                i++;
                if(S1[i] != '\0' && S1[i] != ' ' &&
S1[i] != '\t') return 0;
                else return 1;
            }
        }
    }
}
```

```

    }
return 0;
}
int A ()
{
//this function matches all terminal strings generated by the variable
A. Here the only terminal strings generated by A is ab or a.
    ///??
    if(S1[i] == 'a')
    {
        i++;
        return 1;
    }
    if(S1[i] == 'a')
    {
        i++;
        if(S1[i]=='b')
        {
            i++;
            return 1;
        }
    }
    return 1;
}
else
return 0;
}

```



