**Programs**

**[7] Check Whether Given String Belongs To CFG : S → A010A A → 0A | 1A | ε**

#include<stdio.h>

#include<string.h>

int main()

{

char s[100];

int i,flag,flag1;

int l;

printf("Enter a String : ");

scanf("%s",s);

l=strlen(s);

flag=1;

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

{

if(s[i]!='0' && s[i]!='1')

flag=0;

}

if(flag==1)

printf("string is Valid\n");

else

printf("string is Not Valid\n");

if(flag==1)

{

flag1=0;

for(i=0;i<l-2;i++)

{

if(s[i]=='1')

{

if(s[i+1]=='0' && s[i+2]=='1')

{

flag1=1;

printf("Substring 101 Exists. String Accepted\n");

break;

}

}

}

if(flag1==0)

printf("Substring 101 Doesn't Exist. String Not Accepted\n");

}

}

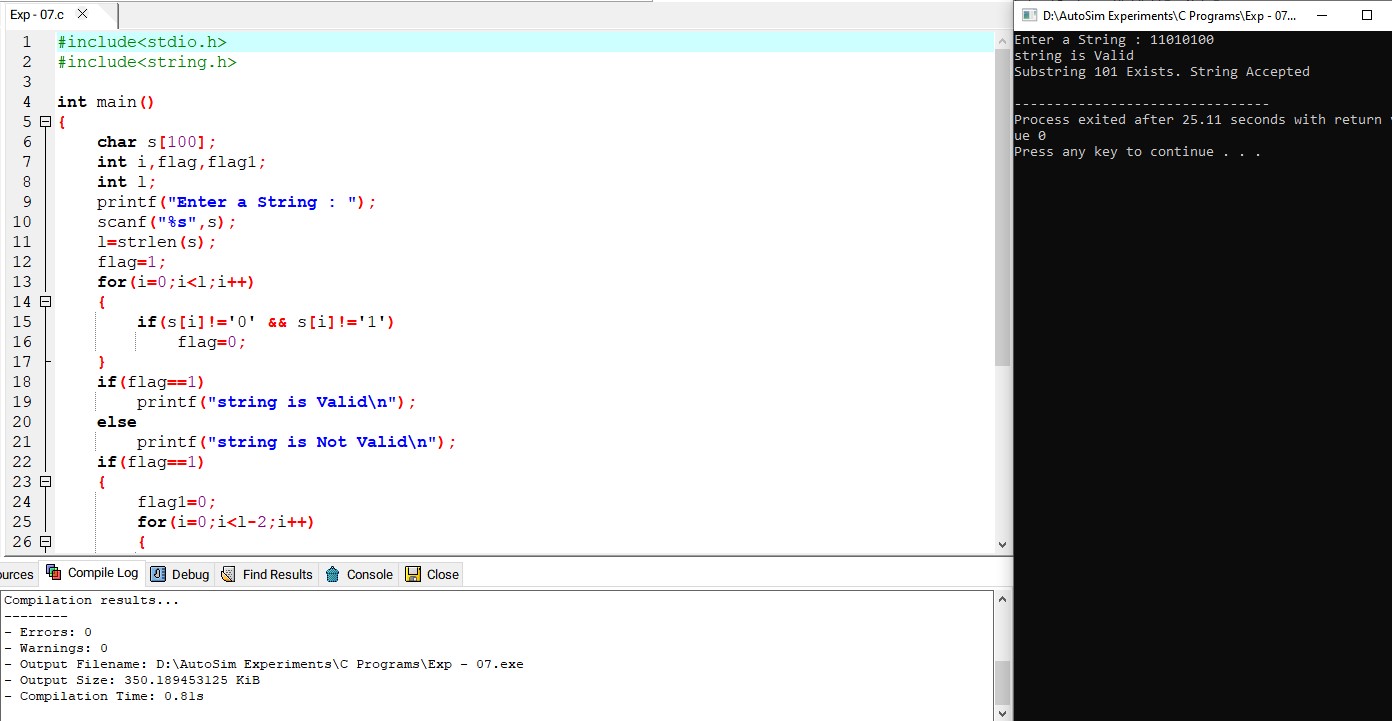
**Algorithm**

**Initialization :** Create a table where each cell [i, j] represents a substring from position i to position j in the input string.

**Fill Terminals :** Populate cells [i, i] with non-terminals that generate the corresponding input character at position i.

**Fill Non-Terminals :** For each substring length len from 2 to input length, iterate over i and partition substring into [i, i+k] and [i+k+1, i+len]. Check non-terminals if they generate these substrings, then add them to cell [i, i+len].

**Acceptance :** If start symbol exists in cell [0, n-1], where n is input length, the input string belongs to the CFG.



**Result**

Thus we have successfully implemented and executed the program and the strings given as inputs are verified