**AIM :**

To write a C program to check whether a string belongs to the grammar

S -> 0 S 0 | 1 S 1 | 0 | 1 | ε

**Language defined by the Grammar**

Set of all strings over 𝚺={0,1} that are palindrome

**ALGORITHM :**

1. Get the input string from the user.

2. Find the length of the string. Let it be n.

3. Check whether all the symbols in the input are either 0 or 1. If so,

print “String is valid” and go to step 4. Otherwise print “String not

valid” and quit the program.

4. If the 1st symbol and nth symbol are the same, 2nd symbol and (n-1)th

symbol are the same and so on, then the given string is palindrome.

So, print “String accepted”. Otherwise, print “String not accepted”

**PROGRAM :**

#include<stdio.h>

#include<string.h>

void main()

{

char s[100];

int i,flag,flag1,a,b;

int l;

printf("enter a string to check:");

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 Not Valid\n");

if(flag==1)

{

flag1=1;

a=0;b=l-1;

while(a!=(l/2))

{

if(s[a]!=s[b])

{

flag1=0;

}

a=a+1;

b=b-1;

}

if (flag1==1)

{

printf("The string is a palindrome\n");

printf("string is accepted\n");

}

else

{

printf("The string is not a palindrome\n");

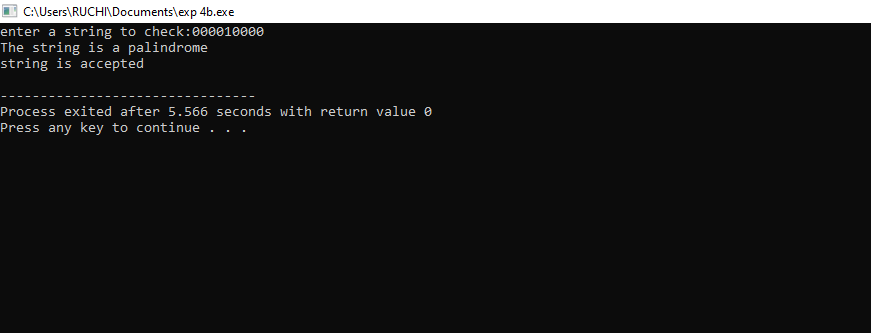
printf("string is Not accepted\n");

}

}

}

**OUTPUT :**



RESULT:

Thus executed successfully