SIMULATING TURING MACHINE

7.To write a C program to simulate a Turing Machine for the language L={ 0n

1n2n | n>=1 } in which n number of 0’s are followed by 2n number of 1’s

PROGRAM :

#include<stdio.h>

#include<string.h>

void main()

{

int i,j,le,flag,flag1,flag2;

char str[20];

printf("Program to show how a turing machine will process 0n1n2n\n");

printf("Enter a string : ");

scanf("%s",str);

le=strlen(str);

j=0;

while(1)

{

flag=0;flag1=0;flag2=0;i=0;

while(i<le)

{

if((str[i]=='0')&&(flag==0))

{

str[i] = 'A';

printf("%s\n",str);

flag=1; //To mark that a 0 is changed to A

i=i+1;

}

else if((str[i]=='0')&&(flag==1))

{

i=i+1; //Skip 0

}

else if(str[i]=='A')

{

i=i+1; //Skip A

}

else if((str[i]=='1')&&(flag1==0))

{

str[i] = 'B';

printf("%s\n",str);

flag1=1; //To mark that a 1 is changed to B

i=i+1;

}

else if((str[i]=='1')&&(flag1==1))

{

i=i+1; //Skip 1

}

else if(str[i]=='B')

{

i=i+1; //Skip B

}

else if((str[i]=='2')&&(flag2==0))

{

str[i] ='C';

printf("%s\n",str);

flag2=1; //To mark that a 2 is changed to C

i=i+1;

}

else if((str[i]=='2')&&(flag2==1))

{

i=i+1; //Skip 2

}

else if(str[i]=='C')

{

i=i+1; //Skip C

}

}

j=j+1;

if(j==le)

{

break;

}

}

}

OUTPUT:

