Write a C program to implement a DFA accepting binary strings ending with ‘00’.

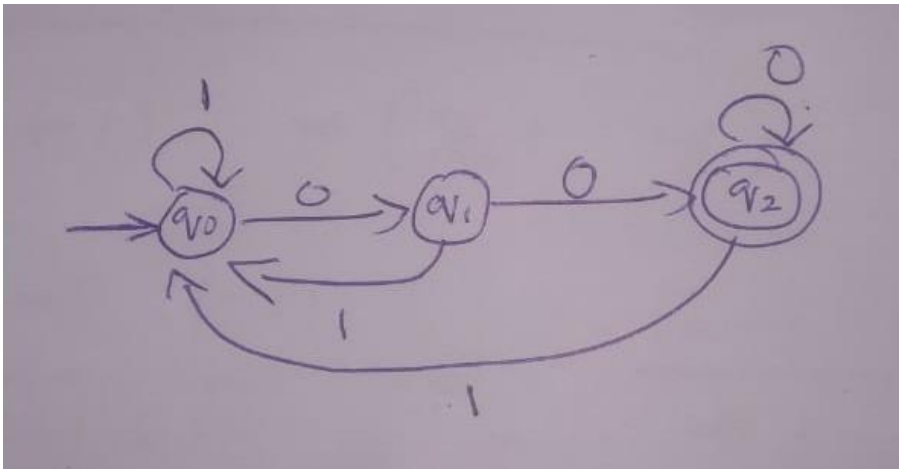
ALGORITHM

Input: A string with characters ‘0’ and ‘1’
Output: Whether the given string is valid or not.

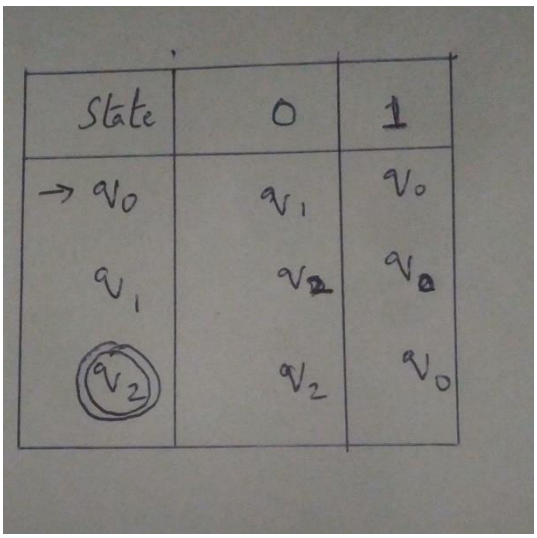
Data structures: Array

Step 1: Start
Step 2: Initialize a character array and input the string.
Step 3: Declare two variables state and i.
Step 4: while input[i]! =’\0’, do
    4a) State is changed according to the current state and input[i]
    4b) if no condition is true then state=’3’
    4c) increment i.
Step 5: If state is equal to final state, print valid string. Otherwise print invalid string.
Step 6: Stop

DFA



Transition Table



State	0	1
→ q_0	q_1	q_0
q_1	q_2	q_0
(q_2)	q_2	q_0

OUTPUT

```
PS D:\Coding\c\Albert Augustine> cd "d:\Coding\c\Albert Augustine\" ; if ($?) { gcc 04.c -o 04 } ; if ($?) { .\04 }
Enter the input string :
00
VALID STRING

PS D:\Coding\c\Albert Augustine> cd "d:\Coding\c\Albert Augustine\" ; if ($?) { gcc 04.c -o 04 } ; if ($?) { .\04 }
Enter the input string :
1100010
INVALID STRING

PS D:\Coding\c\Albert Augustine> cd "d:\Coding\c\Albert Augustine\" ; if ($?) { gcc 04.c -o 04 } ; if ($?) { .\04 }
Enter the input string :
ab0000
INVALID STRING

PS D:\Coding\c\Albert Augustine> cd "d:\Coding\c\Albert Augustine\" ; if ($?) { gcc 04.c -o 04 } ; if ($?) { .\04 }
Enter the input string :
111000
VALID STRING
```