4. Write a C program to check whether a given string belongs to the language defined by a Context Free Grammar (CFG)

S → 0S0 | 1S1 | 0 | 1 | ε

Aim: To write a C program to check whether a given string belongs to the language defined by a Context Free Grammar (CFG)

S → 0S0 | 1S1 | 0 | 1 | ε

Algorithm:

* Read a string from the user.
* Define a recursive function **isBelongToCFG**:
* If the start index exceeds the end index, return 1 (empty string).
* Check rules recursively:

Rule 1: If '0' at both ends, check inside.

Rule 2: If '1' at both ends, check inside.

Rule 3: If '0' at both ends, return 1.

Rule 4: If '1' at both ends, return 1.

* If none matches, return 0.
* Define a function **belongsToCFG** that calls **isBelongToCFG** for the entire string.
* In the main function:
* Get user input.
* Print if the input string belongs to the CFG.

Program:

#include <stdio.h>

#include <string.h>

int isBelongToCFG(char str[], int start, int end) {

if (start > end) {

return 1;

}

if (str[start] == '0' && str[end] == '0') {

return isBelongToCFG(str, start + 1, end - 1);

}

if (str[start] == '1' && str[end] == '1') {

return isBelongToCFG(str, start + 1, end - 1);

}

return 0;

}

int belongsToCFG(char str[]) {

int len = strlen(str);

return isBelongToCFG(str, 0, len - 1);

}

int main() {

char inputString[100];

printf("Enter a string: ");

scanf("%s", inputString);

if (belongsToCFG(inputString)) {

printf("The string belongs to the CFG.\n");

} else {

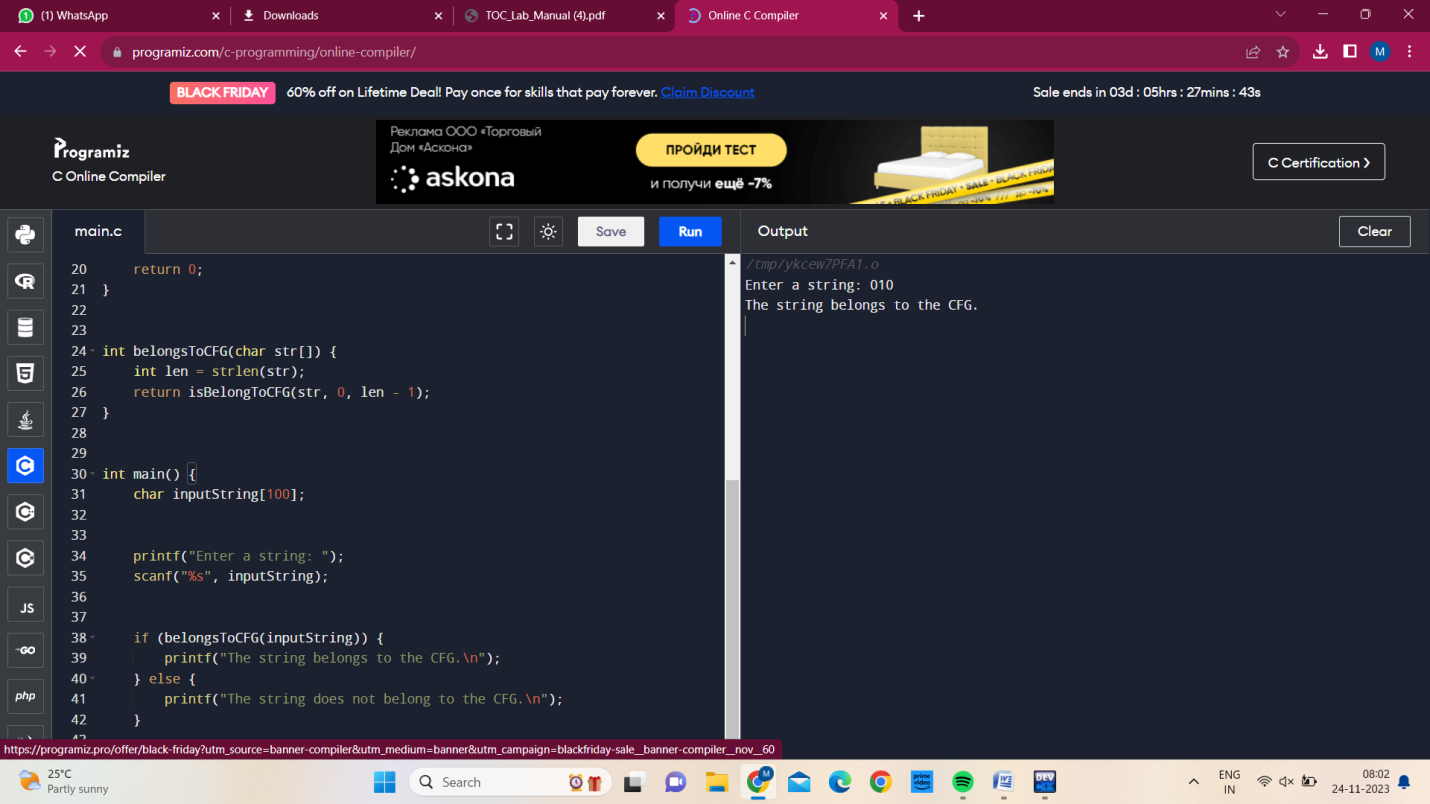
printf("The string does not belong to the CFG.\n");

}

return 0;

}

Output:



Result:

C program has been successfully executed.