* Theory

String:

String is defined as an array of characters or a pointer to characters.

Null - terminated string:

string is terminated by a special characters which is called as Null terminator or Null parameter (10). So when you define a string you should be sure to have sufficient space for the null terminator. The null has value 0

Declaring string:

As in string defination, we have two ways to declare a string. The first way is we declare an array of character

```
char s[] = "string"
```

or

```
char str[20];
```

String operation

To display word with the longest length

This operation involves splitting a string into indivisual words and then determining which word has the maximum length

```
string = "The quick brown fox jumpeed over? the lazy
    print (longest_klord (string))
# output : " Jumpeed ".
```

e) To count the occurrence of each word in a given string

→ This operator involves spliting the string into words and them counting how many times each word appears typically asking a dictionary or counter from the collection module

```
string = " hello hello world"
print (word_count (string))
# output: { 2: hello . 1: world}
```

* algorithm

a) step 1) Initialize maxlength to 0 and longest word to an empty string
step 2) Split the input string into words
step 3) For each word
If its length is greater than maxlength update maxlength and longest word
step 4) Display longest word.

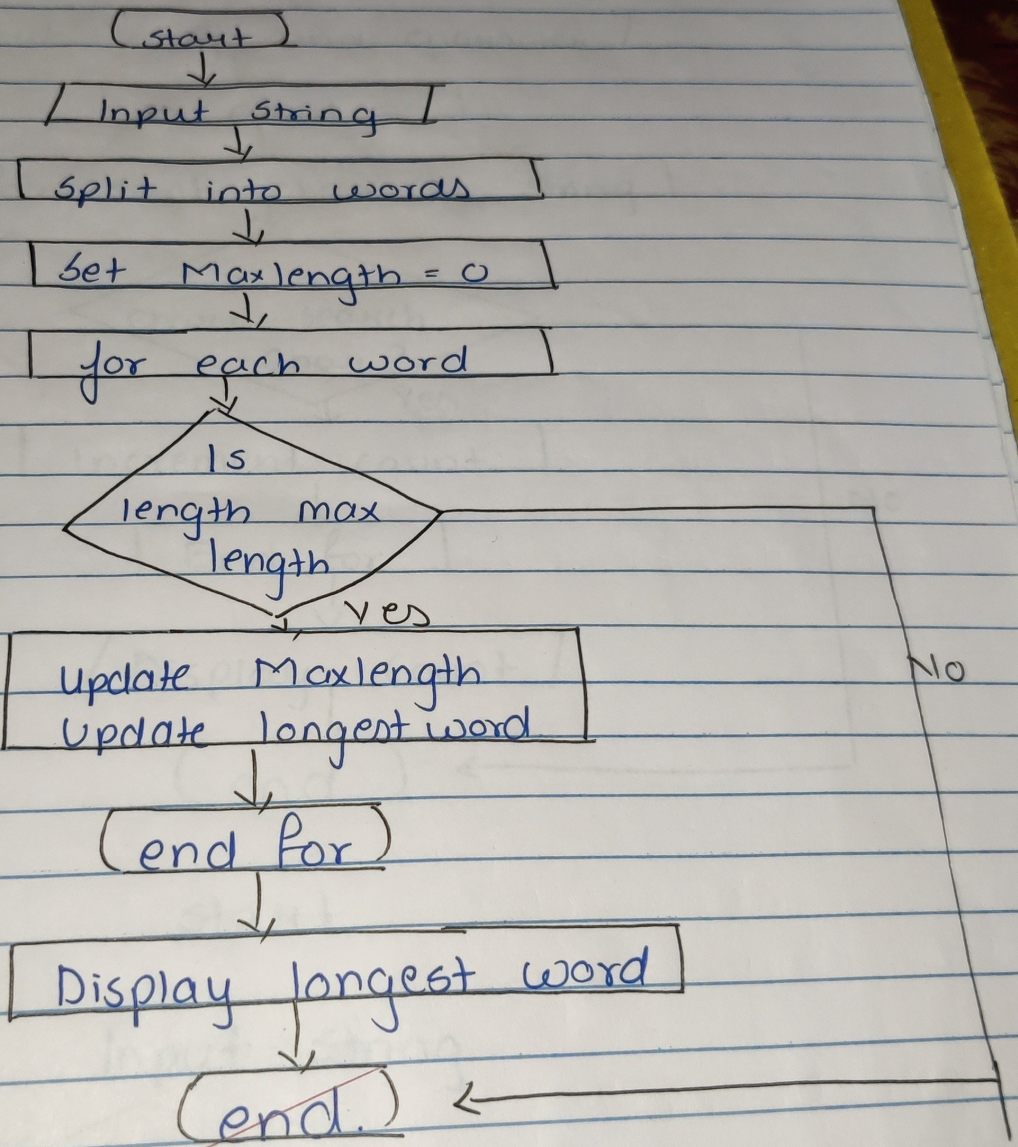
b) step 1) start
step 2) Initialize count to 0
step 3) Input the string and the character to search for
step 4): for each character in the string If it matches the search character increment count
step 5) Display count
step 6) end.

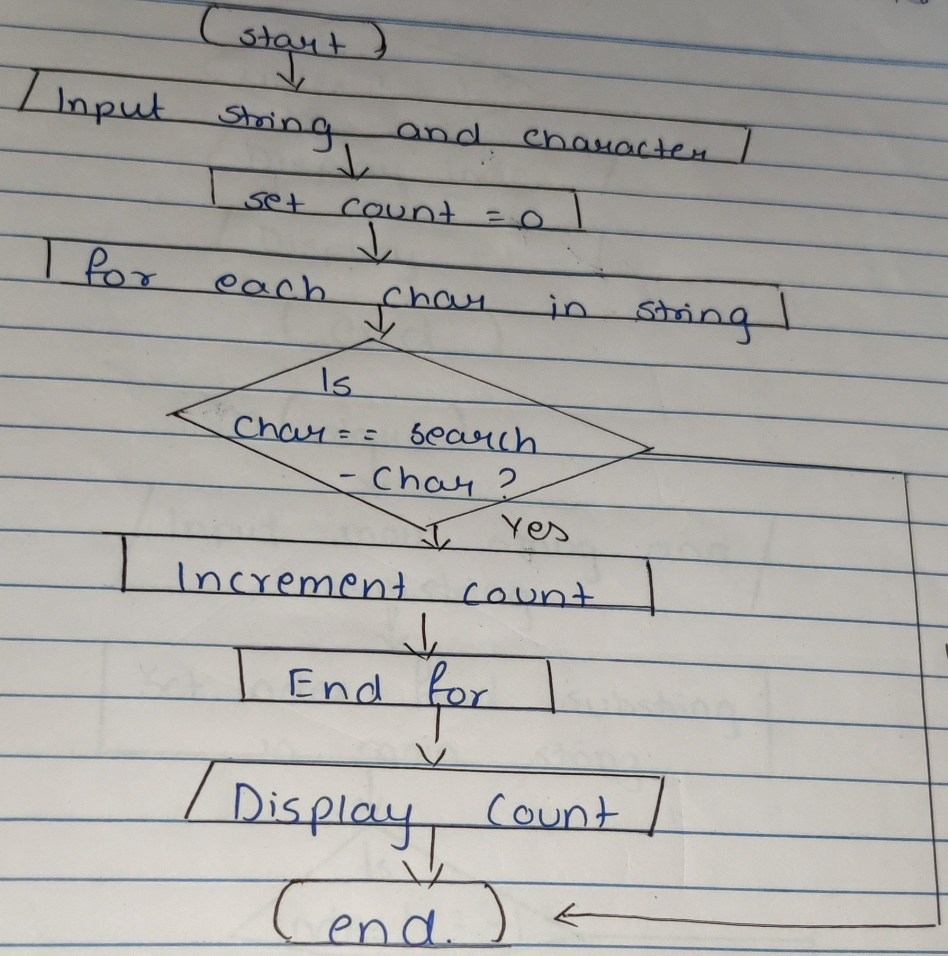
c) step 1) start
step 2) Input the string
step 3) Normalize the string (remove spaces and convert to lowercase)
step 4) reverse the string
step 5) if the normalized string equal the reversed string it is a palindrome
step 6) Display the result
step 7) end.

d) step 1) start
step 2) Input the main string and substring
step 3) find the index of substring in the main string
step 4) if found. display the index otherwise display -1
step 5) end.

e) step 1) start
step 2) Input the string
step 3) split the string into words
step 4) Create a dictionary for word counts
step 5) for each word:
- if it's the dictionary increment it's count
- otherwise add it with a count of 1
step 6) Display the word counts
step 7) end.

Flowchart



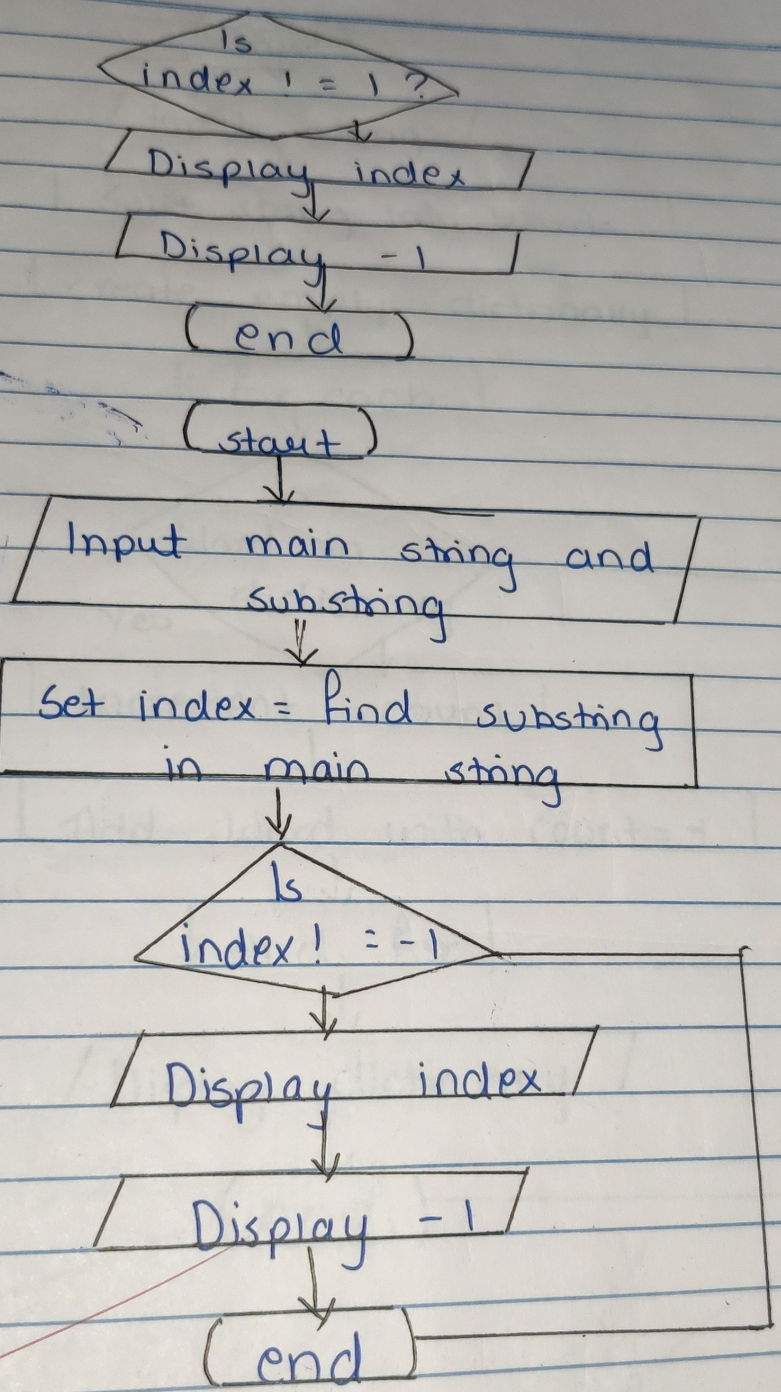



start

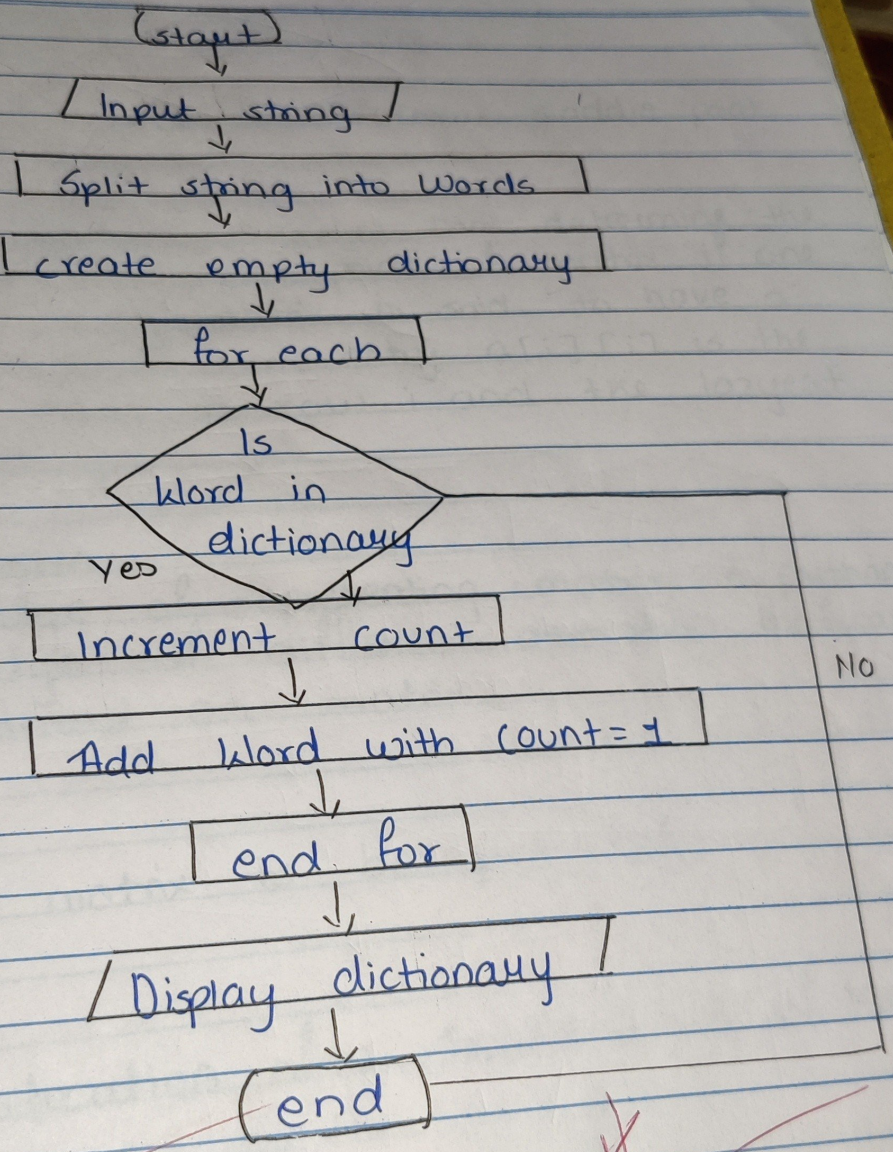
Input string

Normalize string

set reversed string =
reverse (normalize string)



c)




* Conclusion

→ By this way we can perform string operations successfully

12/11/24
7:15 PM