EXERCISE 1

- Input a text in the console.
- Check if the text contains only sorted digits (from lowest to highest values)
- If so, write SORTED, otherwise write NOT SORTED

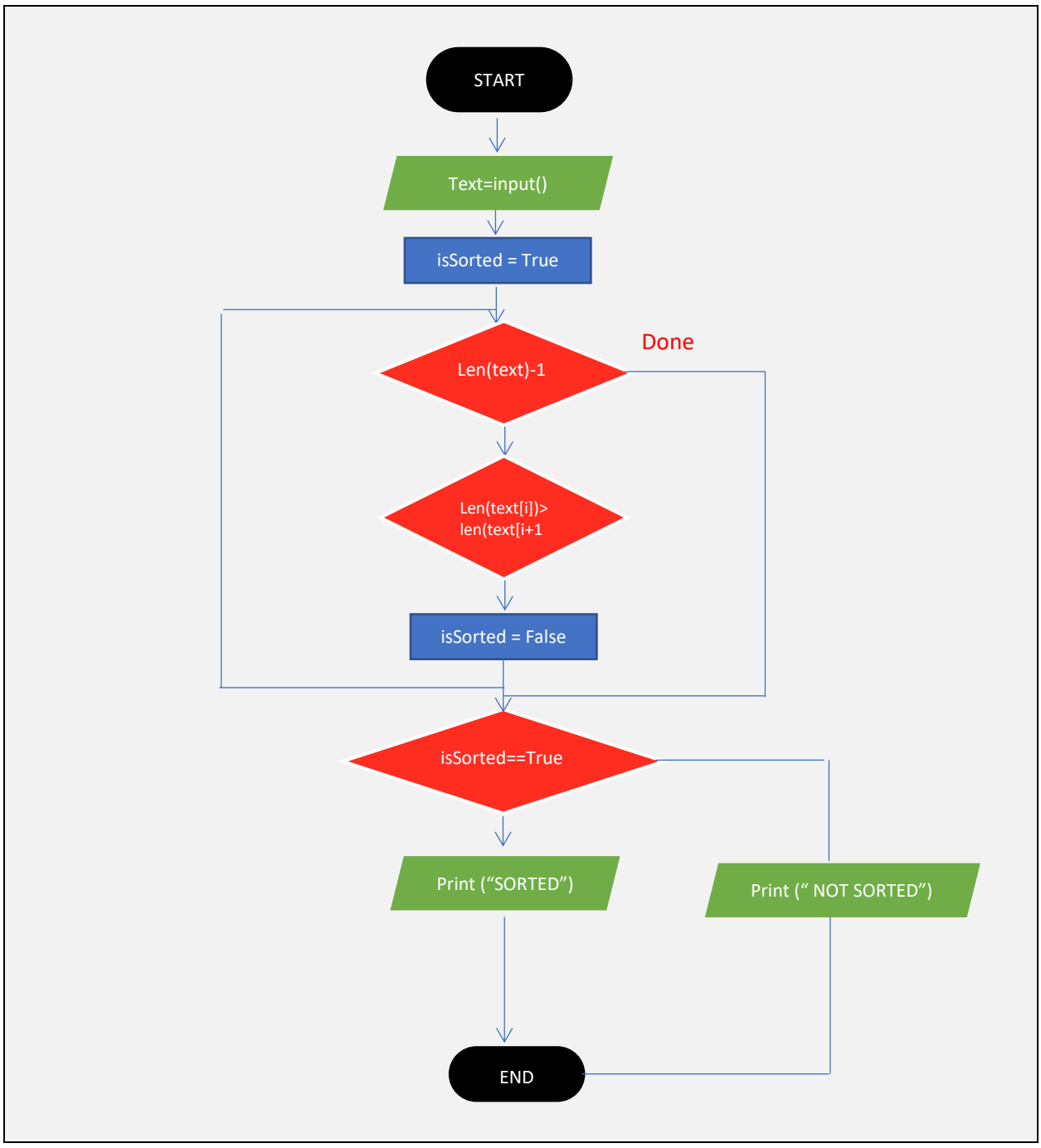
Q1: What will be the **result** for these outputs?

Input	Output
489	SORTED
4762	NOT SORTED
12	SORTED
1268	SORTED
1896	NOT SORTED
1536	NOT SORTED
2789	SORTED

Q2: How many parts can you divide the problem into? Individual work.

Step1: create variable for user input number
Step2 : create isStored for store False
Step3: use for loop to repeat
Step4: check number user in put is order or not
Step5: if number oder print(NTORED)
Step6: if number not order print(NOT STORED)

Q3: Create the flowchart structure of your algorithm. Team (3 students) work.



Q4: Implement your code. Team (3 students) work.

```
text=input()
isSorted = True
for i in range (len(text)-1):
   if text[i] > text[i+1]:
      isSored=False
if isSored == True:
   print("SORTED")
else:
   print("NOT SORTED")
```

Q5: Execute it in a table of execution. Team (3 students) work.

step	text	isSorted	i	Len(text)-1	Text[i]>text[i+1]	Print()
1	489					
2		True				
3			0			
4				2		
5					(4>8)?	
6		True				
						Sorted

EXERCISE 2

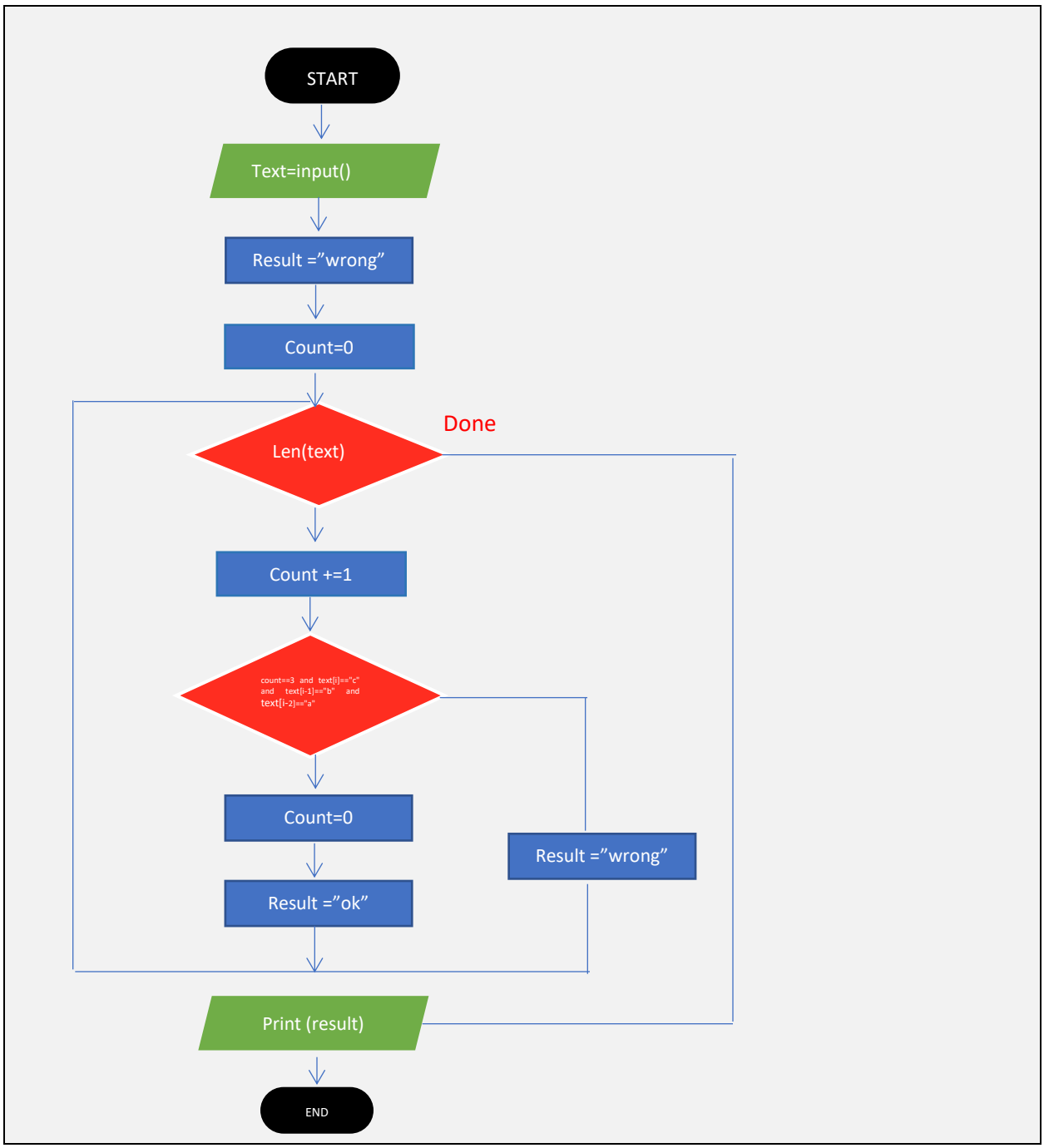
- Input a text in the console.
- Control that the text is owning only "abc" pattern.
    - Print “OK” if so
    - Otherwise, print “WRONG”

Q1: What will be the **result** for these outputs?

Input	Output
abcd	WRONG
abcabc	OK
abc	OK
aabc	WRONG
abbc	WRONG
abcabcab	WRONG
abcdefg	WRONG

Q2: Create your flowchart structure with black boxes.

- Each student has to create his own.
- Share the result in group of 3.



Q3: Implement it in Python. In group of 3.

```
text = input()
result = "WRONG"
count = 0
for i in range(len(text)):
   count += 1
   if count==3 and text[i]=="c" and text[i-1]=="b" and text[i-2]=="a":
      count = 0
      res = "OK"
   else:
      result = "WORNG"
print(res)
```

Q4: Fill up the execution table. In group of 3.

Step	text	result	count	i	Len(text)	Count==3 and text[i]==”c” and text[i-1] ==”b”and text[i-2] ==”a”	Print()
1	abc						
2		Wrong					
3			1				
4				0			
5					3		
6			2				
7				1			
8			3				
9				2			
10						(Count==3)? and (text[2]==c)?and (text[1]==b)? and (text[0]== a)?	
11			0				
12		ok					
13							ok

Q5: Present your flowchart structure to the class. In group of 3.

EXERCISE 3

- Input a text in the console.
- Check that the text:
    - Has only *y*, between square brackets (need open AND close brackets).
    - Otherwise has *x*
- If the text is correct
    - Print “OK”
    - Otherwise, print “WRONG”

Q1: What will be the **result** for these outputs? Individual work

Input	Output
xxx[yyy]xxx	Ok
[yyy]xxx	OK
xxx[yyy	WRONG
xxxy	WRONG
[yy]	WRONG
xxx[yxyy]xxx	WRONG
xxxxx	OK

Q2: Which main instruction can solve the problem? What will it be used for? Group of 3 students.

Step 1: create variable for user input()
Step2 : create variable and set to “”
Step 3 : create variable result for store result change
Step4 : use for loop for repeat
Step 5: use if for check condition if the text have x
Step6 : use elif for check condition if the text have y in [
Step7 : use elif for check condition if the text have y in]
Step8: print()

Q3: Create a code to solve this problem. Group of 3 students.

```
text = input()
is_wrong_display =""
result =''
for i in range(len(text)):
    if text[i] =='x':
        result = 'OK'
    elif i+1<len(text) and text[i] == '[' and text[i+1] == 'y':
        result ='OK'
    elif i+1<len(text) and text[i] == 'y' and (text[i+1] ==']' or text[i+1] =='y') and i !=0 and text[i-1] !='x':
        result ='OK'
    elif text[i] ==']' and text[i-1] =='y':
        result ='OK'
    else:
        is_wrong_display = True
if is_wrong_display == True:
    print("WRONG")
else:
    print(result)
```

Q4: Present your solution to the class. Group of 3 students.