



Rank: 41334 | Points: 1365 (!)

Group(), Groups() & Groupdict()



X Your Group(), Groups() & Groupdict() submission got 20.00 points. Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial A

group()

A group() expression returns one or more subgroups of the match.

Code

```
>>> import re
>>> m = re.match(r'(\w+)@(\w+)\.(\w+)','username@hackerrank.com')
                  # The entire match
>>> m.group(0)
'username@hackerrank.com'
>>> m.group(1)  # The first parenthesized subgroup.
'username'
               # The second parenthesized subgroup.
>>> m.group(2)
'hackerrank'
>>> m.group(3)
                  # The third parenthesized subgroup.
>>> m.group(1,2,3) # Multiple arguments give us a tuple.
('username', 'hackerrank', 'com')
```

groups()

A groups() expression returns a tuple containing all the subgroups of the match.

Code

```
>>> import re
>>> m = re.match(r'(\w+)@(\w+)\.(\w+)','username@hackerrank.com')
>>> m.groups()
('username', 'hackerrank', 'com')
```

groupdict()

A groupdict() expression returns a dictionary containing all the named subgroups of the match, keyed by the subgroup name.

Code

```
>>> m = re.match(r'(?P<user>\setminus w+)@(?P<website>\setminus w+)\setminus.(?P<extension>\setminus w+)','myname@hackerrank.com')
{'website': 'hackerrank', 'user': 'myname', 'extension': 'com'}
```

Task

You are given a string S.

Your task is to find the first occurrence of an alphanumeric character in S (read from left to right) that has consecutive repetitions.

Input Format

A single line of input containing the string S.

Constraints

 $0 < \operatorname{len}(S) < 100$

Output Format

Print the first occurrence of the repeating character. If there are no repeating characters, print -1.

Sample Input

..12345678910111213141516171820212223

Sample Output

1

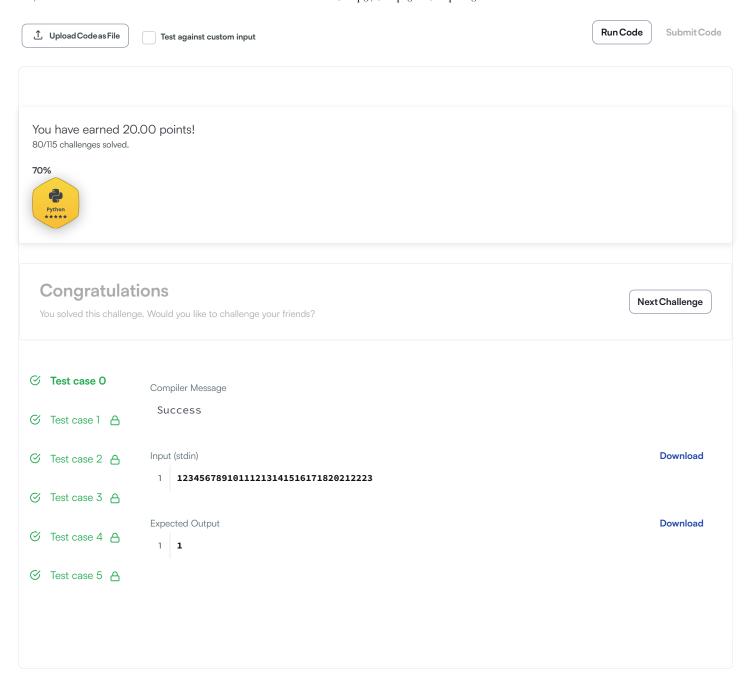
Explanation

- .. is the first repeating character, but it is not alphanumeric.
- 1 is the first (from left to right) alphanumeric repeating character of the string in the substring 111.

```
Change Theme Language Python 3
    # Enter your code here. Read input from STDIN. Print output to STDOUT
 1
    import re
 2
 3
 4
    pattern = r''([a-zA-Z0-9])\1"
 5
    s = input()
 6
    m = re.search(pattern, s)
 7
    if m:
         print(m.group(1))
 8
 9
    else:
         print(-1)
10
11
12
```

EMACS Line: 12 Col: 1

૱ :



Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy