

Лабораторна робота №2 (python-2)

Процедури користувача. Обробка символьних рядків.

У наведеному нижче завдання необхідно:

Написати функції користувача мовою програмування Python для виконання завдання, зазначеного у відповідному варіанті.

Обов'язково записати інструкцію щодо використання розробленої процедури/функції та навести приклад її використання (головну тестову програму). Тестова програма повинна вводити з консолі цей текстовий рядок.

Примітка: Виконати необхідні дії ДВОМА способам:

- використовувати БАЗОВІ операції з символами та рядками;

- використовувати методи відповідних об'єктів програми.

1. Даний текст. З'ясувати процентний зміст у тексті символів «#» та «@».
2. Даний текст. Чи з'ясувати він є паліндром. (Див., наприклад wikipedia.org). Примітка перед перевіркою, рекомендується викинути з тексту символи ПРОБІЛ.
3. Даний текст містить слова, розділені кількома пробілами. Отримати текст, який містить слова, розділені ОДНИМ пробілом. [Дивись тут.](#)
4. Даний текст. Отримати рядок, у якому відсутні пробіли, розташовані на початку рядка.

5. Даний текст. Виконати просте шифрування тексту. Кожен символ, код якого є x , замінити символом із кодом $F(x)$.
Вибрати $F(x) = 32767 - x$ Тестова програма повинна виконати перекодування вихідного рядка на кодований, потім отриманий кодований рядок «закодувати» ще раз. Порівняйте та поясніть отриманий результат.
6. Даний текст. З'ясувати процентний зміст у тексті літери a та A
7. Даний текст. З'ясувати відсотковий зміст у тексті заданого слова.
8. Даний текст. З'ясувати відсотковий зміст у тексті заданої літери.
9. Даний текст. З'ясувати відсотковий зміст у тексті «здвоєних» прогалін.
10. Даний текст. З'ясувати процентний зміст у тексті символів «%» та «*»
11. Даний текст. З'ясувати відсотковий зміст у тексті символів «&» та «?»
12. Даний текст. З'ясувати процентний зміст у тексті символів «@» та «!»
13. Даний текст. З'ясувати відсотковий зміст у тексті символу «мінус» та «(»
14. Даний текст. З'ясувати відсотковий зміст у тексті символу "(" і ")"
15. Даний текст. З'ясувати відсотковий зміст у тексті символів "цифра в десятковій системі числення"
16. Даний текст. Отримати рядок, у якому відсутні прогаліни, розташовані в кінці рядка.
17. Даний текст. З'ясувати, чи цей текст є ідентифікатором змінної;
18. Даний текст. З'ясувати, чи є текст текстовим записом цілого числа.
19. Даний текст. Групи символів, розділені пробілами (одним або декількома) і не містять пробілів у собі, називатимемо словами. ([Дивись тут.](#)) Випередити кількість слів у тексті.

20. Даний текст. Якщо в тексті немає символу *, залишити цей текст без зміни, інакше кожен з англійських літер, що передують першому входженню символу *, замінити на символ #.
21. Даний текст. Якщо в тексті немає символу +, то залишити текст без зміни, інакше кожен цифру, що передує першому входженню символу +, замінити символом @.
22. Даний текст. Замінити у тексті усі літери Я літерою Q.
23. Даний текст. Замінити у тексті всі літери англійського алфавіту символом ТІЛЬДА.
24. Даний текст. Замінити у тексті всі символи «цифра» символом #
25. Даний текст. Замінити у тексті всі ЗАДАННІ у вхідному рядку символи символом &
26. Даний текст. Замінити текст Задане слово ([Дивись тут](#)) на символ *
27. Даний текст. Замінити в тексті Задане(1) слово на задане(2) слово ([Дивись тут](#))
28. Даний текст. Кожне слово тексту записати у дзеркальному вигляді ([Дивись тут](#))
29. Даний текст. Знайти скільки разів у тексті зустрічається задане поєднання букв
30. Даний текст; визначити, чи містить він символи, відмінні від англійських літер та пробілу.

Приклад рішення

```
# Даний текст. Замінити в тексті всі ЗАДАНІ
# у вхідному рядку символи символом &.
def schange(text_in, str_change):
```

```
"""
```

```
У текстовому рядку text_in замінюються всі  
символи, записані в заданому рядку str_change  
на символ &
```

```
"""
```

```
str=""
```

```
for st in text_in:  
    out_str=st  
    for ch in str_change:  
        if ch==out_str:  
            out_str='&'  
            break  
    str+=out_str  
return str
```

```
#####
```

```
ch=input("введіть символи, що замінюються в рядку: ")
```

```
while True:
```

```
    txt=input("введіть текст (Enter - stop pgm):\n")
```

```
    if len(txt)==0:
```

```
        break
    txt1 = schange (txt, ch)
    print("old txt=",txt)
    print("new txt=",txt1)
```

```
введіть символи, що замінюються в рядку: ЁoiIїi5
введіть текст (Enter - stop pgm):
eoooo444єєіііііііі7777
old txt=
new txt= &&&&444ЄЄЄ&&&&&&&7777
введіть текст (Enter - stop pgm):
```

Приклад отримання списку слів, поділених заданими роздільниками

```
# з вхідного рядка s отримуємо list слів,
# розділених символами роздільниками з рядка SEPARATORS

def toster_word_split(s, SEPARATORS=' \n\r\t.;;?!\\t'):
    result=[]
```

```
current_word=''
for char in s:
    if char in SEPARATORS:
        if current_word:
            result.append(current_word)
            current_word=''
        else:
            current_word+=char

    if current_word:
        result.append(current_word)
return result
```

```
txt=' 111 222 3333... ;;;43444 555 2222....@'
w1=toster_word_split(txt)
print(w1)
```

ВЫВОД:

```
['111', '222', '3333', '43444', '555', '2222', '@']
```