

Практическое задание № 7

Тема: составление программ со строками в IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со строками в IDE PyCharm Community.

Постановка задачи:

Задача № 1: Дано целое число N ($32 < N < 126$). Вывести символ с кодом, равным N .

Текст программы:

```
# Дано целое число N (32 < N < 126). Вывести символ с кодом, равным N.  
import random  
  
N = random.randrange(32, 126)  
print("N = ", N, "; Char: ", chr(N))
```

Протокол программы:

N = 92 ; Char: \

Process finished with exit code 0

Тип алгоритма: линейный

Задача № 2: Дана строка, содержащая латинские буквы и круглые скобки. Если скобки расставлены правильно (то есть каждой открывающей соответствует одна закрывающая), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная закрывающая скобка, или, если закрывающих скобок не хватает, число —1.

Текст программы:

```
# Дана строка, содержащая латинские буквы и круглые скобки. Если скобки  
# расставлены правильно (то есть каждой открывающей
```

```

соответствует одна
# закрывающая), то вывести число 0. В противном случае вывести
или номер позиции,
# в которой расположена первая ошибочная закрывающая скобка,
или, если
# закрывающих скобок не хватает, число -1.
def check_brackets(s):
    # Инициализация пустого стека
    List = []

    # Итерация по символам строки с учётом их позиции (start=1)
    for i, char in enumerate(s, start=1):
        # Если символ - открывающая скобка, добавляем её в стек
        # вместе с позицией
        if char == '(':
            stack.append((char, i))
        # Если символ - закрывающая скобка
        elif char == ')':
            # Если лист пуст, возвращаем позицию ошибочной
            # закрывающей скобки
            if not List:
                return i # Закрывающая скобка без пары
            # Иначе удаляем последнюю открывающую скобку из
            # стека
            stack.pop()

    # После завершения цикла проверяем, остались ли
    # несогласованные открывающие скобки
    if not List:
        return 0 # Все скобки согласованы
    else:
        # Если остались, возвращаем позицию первой открывающей
        # скобки без закрывающей
        return List[-1][1] # Первая открывающая скобка без
        # закрывающей

# Пример использования
input_string = "a(b)c)d"
result = check_brackets(input_string)

# Вывод результата
if result == 0:
    print("Скобки расставлены правильно.")
elif result == -1:
    print("Недостаточно закрывающих скобок.")
else:
    print(f"Ошибка на позиции {result}.")

```

Протокол программы:

Ошибка на позиции: 6.

Process finished with exit code 0

Тип алгоритма: циклический, ветвление

Вывод: В ходе работы закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрёл навыки составления программ со строками в IDE PyCharm Community.