

Практическая работа № 7

Тема: Составление программ со строками в IDE PC Community.

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

Постановка задачи: Дано целое число N ($32 < N < 126$). Вывести символ с кодом, равным N .

Тип алгоритма: Линейный

Текст программы:

```
## Дано целое число N (32 < N < 126). Вывести символ с кодом, равным N.

try:
    # Чтение целого числа N
    N = int(input('Введите число: '))

    # Проверка условия 32 < N < 126
    if not (32 < N < 126):
        print("Число должно быть в диапазоне от 33 до 125")
    else:
        # Преобразование числа в символ
        symbol = chr(N)
        print(symbol)

except ValueError:
    print("Вы ввели неверный формат данных. Ожидается целое число.")
```

Протокол работы программы:

Введите число: 34
"

Process finished with exit code 0

Постановка задачи: 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.

Тип алгоритма: Лине́йный

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): 1usage  ▲NiceKirill
    stack = [] # Стек для хранения позиций открывающих скобок
    for i, char in enumerate(s):
        if char == '(':
            stack.append(i) # Помещаем индекс открывающей скобки в стек
        elif char == ')':
            if not stack: # Если стек пуст, значит, закрывающая скобка без пары
                return i # Возврат позиции первой ошибочной закрывающей скобки
            stack.pop() # Убираем соответствующую открывающую скобку из стека

    # Если после прохода по строке остались символы в стеке, значит, не хватало закрывающих скобок
    return -1 if stack else 0

# Тестирование функции
tests = [
    "((( )))", # Балансирован
    "((( )))(", # Первая ошибочная закрывающая скобка на позиции 4
    "())", # Не хватает закрывающих скобок
    "(( ))", # Балансирован
    "))))", # Первая ошибочная закрывающая скобка на позиции 0
    "(" # Не хватает закрывающих скобок
]

for test in tests:
    try:
        result = check_brackets(test)
        print(f'Строка "{test}" -> Результат: {result}')

    except ValueError as e:
        print(f'Строка "{test}" -> Ошибка: {e}')
```

Протокол работы программы:

Строка "((()))" -> Результат: 0

Строка "((()))(" -> Результат: 4

Строка "())" -> Результат: 2

Строка "(())" -> Результат: 0

Строка "))))" -> Результат: 0

Строка "(" -> Результат: -1

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ линейной структуры в IDE PyCharm Community.

Были использованы языковые конструкции `try`, `except`, `if not`, `else`, `def`, `raise`, `return`, `if`, `for`, `in`. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.