

## Практическое занятие №17.

**Тема: «составление программ с использованием GUI Tkinter в IDE PyCharm Community, изучение возможностей модуля OS»**

**Цель: «закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучить возможности модуля OS.»**

### Постановка задачи №1.

```
# Разработать программу с применением пакета tk, взяв в качестве условия одну  
# любую задачу из ПЗ №№ 2 - 9.
```

### Тип алгоритма №1: Ветвление

### Код алгоритма №1

```
import tkinter as tk
import tkinter.messagebox as messagebox

def Myfunction():
    try:
        a = int(entry_a.get())
        if a < 0:
            raise ValueError("Введите положительное число")
        b = entry_b.get()

        if not b:
            raise ValueError("Поле пустое")

        c = b * a

        result_text.delete("1.0", "end")
        result_text.insert("1.0", c)

    except ValueError as e:
        messagebox.showerror("Ошибка", e)

root = tk.Tk()
root.title("Повторение символов")

frame = tk.Frame(root)
frame.pack(padx=10, pady=10)

entry_a_label = tk.Label(frame, text="Введите число символов для строки:")
entry_a_label.grid(row=0, column=0, padx=(0, 20), pady=(20, 0))

entry_a = tk.Entry(frame)
```

```

entry_a.grid(row=0, column=1)

entry_b_label = tk.Label(frame, text="Введите символы, которые вы хотите
повторить:")
entry_b_label.grid(row=1, column=0, padx=(0, 20), pady=(10, 0))

entry_b = tk.Entry(frame)
entry_b.grid(row=1, column=1)

button = tk.Button(frame, text="Выполнить", command=Myfunction)
button.grid(row=2, columnspan=2, pady=(20, 20))

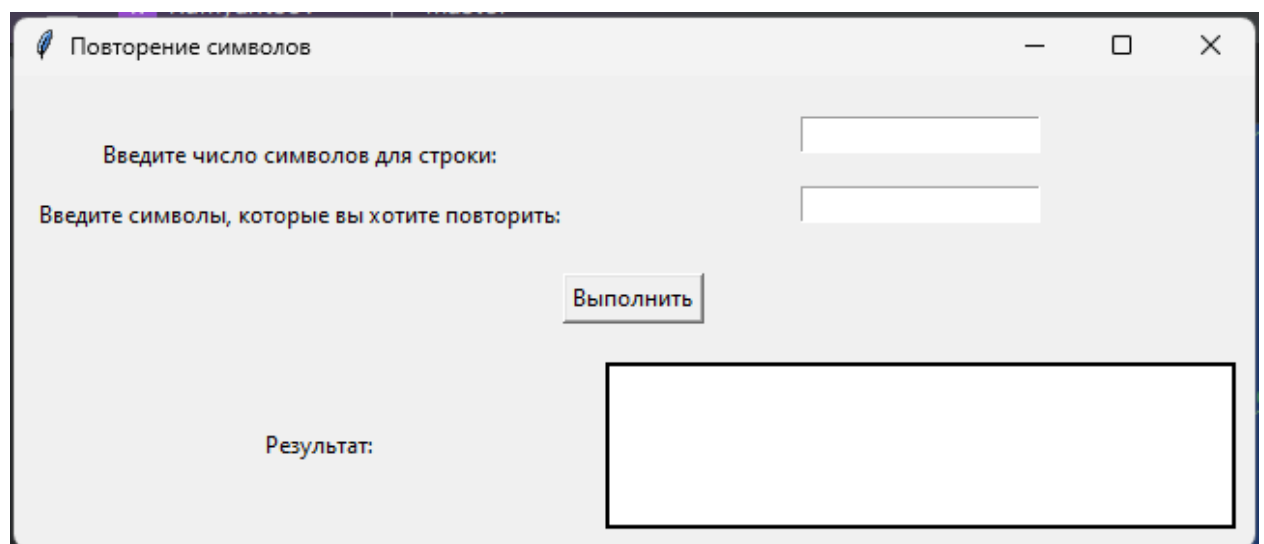
result_label = tk.Label(frame, text="Результат:")
result_label.grid(row=3, column=0)

result_text = tk.Text(frame, width=40, height=5, borderwidth=2,
relief="solid")
result_text.grid(row=3, column=1)

root.mainloop()

```

## Вывод алгоритма:



## Постановка задачи №2.

```

# перейдите в каталог PZ11. Выведите список всех файлов в этом каталоге.
Имена
# вложенных подкаталогов выводить не нужно.
# | перейти в корень проекта, создать папку с именем test. В ней создать еще
одну папку
# test1. В папку test переместить два файла из ПЗ6, а в папку test1 - один
файл из ПЗ7.
# Файл из ПЗ7 переименовать в test.txt. Вывести в консоль информацию о
размере
# файлов в папке test.
# | перейти в папку с PZ11, найти там файл с самым коротким именем, имя
вывести в
# консоль. Использовать функцию basename () (os.path.basename()).
# | перейти в любую папку где есть отчет в формате .pdf и «запустите» файл в

```

```
# привязанной к нему программе. Использовать функцию os.startfile().  
# } удалить файл test.txt.
```

## Тип алгоритма: Линейный

### Код алгоритма №2.

```
import os  
  
print("Задание №1")  
os.chdir("../PZ_11")  
  
files = []  
  
for obj in os.listdir():  
    if os.path.isfile(obj):  
        files.append(obj)  
  
print(files)  
  
print("Задание №2")  
  
os.chdir("../")  
os.mkdir("test")  
os.mkdir("test/test1")  
  
with open("../PZ_6/Практическая №6().pdf", "rb") as src_file:  
    with open("test/Практическая №6().pdf", "wb") as dst_file:  
        dst_file.write(src_file.read())  
  
with open("../PZ_6/PZ_6.py", "r", encoding="utf-8") as src_file:  
    with open("test/pz6_ivahnenko.py", "w", encoding="utf-8") as dst_file:  
        dst_file.write(src_file.read())  
  
with open("../PZ_6/PZ_6(2).py", "r", encoding="utf-8") as src_file:  
    with open("test/pz6_ivahnenko.py", "w", encoding="utf-8") as dst_file:  
        dst_file.write(src_file.read())  
  
with open("../PZ_6/PZ_6(3).py", "r", encoding="utf-8") as src_file:  
    with open("test/pz6_ivahnenko.py", "w", encoding="utf-8") as dst_file:  
        dst_file.write(src_file.read())  
  
with open("../PZ_7/PZ_7.py", "r", encoding="utf-8") as src_file:  
    with open("test/test1/test.txt", "w", encoding="utf-8") as dst_file:  
        dst_file.write(src_file.read())  
  
with open("../PZ_7/PZ_7(2).py", "r", encoding="utf-8") as src_file:  
    with open("test/test1/test.txt", "w", encoding="utf-8") as dst_file:  
        dst_file.write(src_file.read())  
  
size = []  
for file in os.listdir("test"):  
    if os.path.isfile(os.path.join("test", file)):  
        size.append(os.path.getsize(os.path.join("test", file)))
```

```

print(size)

print("Задание №3")

os.chdir("./PZ_11")

short_filename = ""
for filename in os.listdir():
    if len(filename) < len(short_filename) or short_filename == "":
        short_filename = filename

print(os.path.basename(short_filename))

print("Задание №4")
print("Файл из PZ_2 открыт")

pdf_folder = '../PZ_2'

pdf_filename = 'Практическое задание№2(2).pdf'

pdf_path = os.path.join(pdf_folder, pdf_filename)

if os.path.isfile(pdf_path):
    os.startfile(pdf_path)
else:
    print("Такого файла не существует")

print("Пункт №5")

os.chdir = '../test/test1'

file_path = os.path.join(os.chdir, 'test.txt')

if os.path.isfile(file_path):
    os.remove(file_path)
    print('Файл успешно удален.')
else:
    print('Файл не найден.')

```

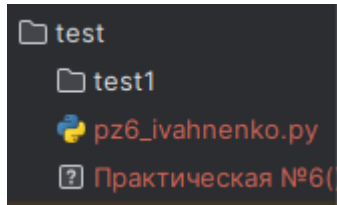
## Вывод алгоритма:

```

Задание №1
['PZ_11(2).py', 'PZ_11.pdf', 'PZ_11.py', 'text18-24(2)', 'text18-24.txt', 'text_file_1.txt', 'text_file_2.txt', 'text_file_3.txt']
Задание №2
[1118, 590603]
Задание №3
PZ_11.py
Задание №4
Файл из PZ_2 открыт
Пункт №5
Файл успешно удален.

```

Что добавляется или изменяется во время запуска алгоритма:



(Открывается файл из ПЗ\_2)

## Практическое задание №2

**Тема:** Знакомство и работа с IDE PyCharm Community.  
Построение программ линейной структуры в IDE PyCharm Community.

**Цель:** Выработка первичных навыков работы с IDE PyCharm Community, составление программ линейной структуры.

### Задача:

Разработка программы, показывающую, сколько полных минут прошло с начала суток.

**Тип алгоритма:** Линейный



**Вывод: «Я закрепил понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с использованием GUI Tkinter в IDE PyCharm Community, изучил возможности модуля OS.»**