|  |  |  |
| --- | --- | --- |
| Защищено:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |  | Демонстрация:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

**Отчет по лабораторной работе №4 по курсу**

**«Парадигмы и конструкции языков программирования»**

#### Тема работы: " **Калькулятор систем счисления** "

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: | Пыжьянов Александр Сергеевич |
| студент группы  ИУ5Ц-52Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| Гапанюк Ю.Е. | "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Москва, МГТУ 2024

**СОДЕРЖАНИЕ**

[Цель лабораторной работы 3](#_Toc176887611)

[Практическое задание 3](#_Toc176887612)

[Листинг программы 4](#_Toc176887613)

[Результат работы программы 8](#_Toc176887614)

## Цель лабораторной работы

Научиться работе с библиотекой tkinter.

## Практическое задание

Составить приложение, используя модуль создания оконных приложений

Tkinter, реализующее Перевод заданного вещественного числа из 10-й системы счисления в 3-ю и обратно.

Интерфейс должен предоставлять ввод символов: как числовых, так и знаков

операций и с использованием клавиатуры, и с помощью кнопок приложения.

Также в приложении необходимо создать меню, в котором должны быть

следующие пункты:

● заданные действия,

● очистка полей ввода/вывода (по одному и всех сразу),

● информация о программе и авторе.

## Листинг программы

**import** tkinter **as** tk

**from** tkinter **import** ttk

**def** visual():

**def** save\_number():

entry\_chislo.config(state = 'readonly')

**def** save\_vibor():

vibor.config(state = 'disabled')

**def** clear\_number():

entry\_chislo.config(state = 'normal')

entry\_chislo.delete('0', 'end')

**def** clear\_vibor():

vibor.config(state = 'readonly')

vibor.delete('0', 'end')

**def** clear\_all():

entry\_chislo.config(state = 'normal')

entry\_chislo.delete('0', 'end')

vibor.config(state = 'readonly')

vibor.delete('0', 'end')

**def** perevod():

vib = vibor.get()

number = entry\_chislo.get()

**if** vib == 'Из 10-й в 3-ю':

ans = ten\_to\_third(number)

**else**:

ans = third\_to\_ten(number)

label\_ans.config(text = ans)

**def** end\_prog():

root.destroy()

**def** add\_from\_box():

entry\_chislo.insert('end' , vibor\_chisla.get())

**def** clear\_vivod():

label\_ans.config(text = '')

root = tk.Tk() #создание окна

root.title('Перевод из 10-й системы в 3-ю и обратно') #заголовок

root.geometry('900x600') #исходный размер

label\_info = tk.Label(text = 'Работу выполнил Александр Пыжьянов \n группа ИУ5Ц-52Б') #создание текстового объекта

label\_info.grid(row = 0, column = 0) #расположение объекта

label\_vvod = tk.Label(text = 'Введите число для перевода')

entry\_chislo = tk.Entry() #создание поля для ввода

chisla = (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, '.')

vibor\_chisla = ttk.Combobox(values = chisla, state = 'readonly') #создание мультивыбора

btn\_vvod = ttk.Button(text = 'Ввод', command = save\_number) #создание кнопки

btn\_vvod\_clear = ttk.Button(text = 'Очистить', command = clear\_number)

btn\_vvod\_box = ttk.Button(text = 'Добавить', command = add\_from\_box)

label\_vvod.grid(row = 1, column = 0, sticky = 'w')

entry\_chislo.grid(row = 1, column = 1, sticky = 'w')

btn\_vvod.grid(row = 1, column = 2, sticky = 'w')

btn\_vvod\_clear.grid(row = 1, column = 3, sticky = 'w')

vibor\_chisla.grid(row = 1, column = 4, sticky = 'w')

btn\_vvod\_box.grid(row = 1, column = 5, sticky = 'w')

label\_vibor = tk.Label(text = 'Выберите, из какой СС в какую перевод')

vibor = ttk.Combobox(values = ('Из 10-й в 3-ю', 'Из 3-й в 10-ю'), state = 'readonly')

btn\_vibor = ttk.Button(text = 'Ввод', command = save\_vibor)

btn\_vibor\_clear = ttk.Button(text = 'Очистить', command = clear\_vibor)

label\_vibor.grid(row = 2, column = 0, sticky = 'w')

vibor.grid(row = 2, column = 1, sticky = 'w')

btn\_vibor.grid(row = 2, column = 2, sticky = 'w')

btn\_vibor\_clear.grid(row = 2, column = 3, sticky = 'w')

btn\_clear\_all = ttk.Button(text = 'Очистить все поля', command = clear\_all)

btn\_clear\_all.grid(row = 3, column = 0, sticky = 'w')

btn\_deyst = ttk.Button(text = 'Перевести число', command = perevod)

btn\_deyst.grid(row = 4, column = 0, sticky = 'w')

label\_rez = tk.Label(text = 'Результат перевода: ')

label\_rez.grid(row = 5, column = 0, sticky = 'w')

label\_ans = tk.Label(text = 'Не было вычислений')

label\_ans.grid(row = 5, column = 1, sticky = 'w')

btn\_end = ttk.Button(text = 'Завершить выполнение программы', command = end\_prog)

btn\_end.grid(row = 6, column = 0, sticky = 'w')

btn\_clear\_vivod = ttk.Button(text = 'Очистить', command = clear\_vivod)

btn\_clear\_vivod.grid(row = 5, column = 2, sticky = 'w')

root.mainloop()

**def** ten\_to\_third(x):

**try**:

x = float(x)

cel = int(x)

drob = x - cel

ans\_cel = ''

ans\_drob = ''

**while** cel > 0:

d = cel % 3

cel //= 3

ans\_cel += str(d)

ans\_cel = ans\_cel[::-1]

k = 0

**while** drob != int(drob) **and** k < 10:

drob = drob - int(drob)

drob \*= 3

tec = int(drob)

ans\_drob += str(tec)

k += 1

**if** drob != 0:

ans\_drob += str(int(drob))

**if** drob == 0:

ans = ans\_cel

**else**:

ans = ans\_cel + '.' + ans\_drob

**except**:

ans = 'Введено некорректное число'

**return** ans

**def** third\_to\_ten(x):

**try**:

test = float(x)

ans = 0

f = **True**

bad = '3456789'

**if** '.' **in** x:

tochka = x.index('.')

d = len(x) - (len(x) - tochka) - 1

**for** i **in** x:

**if** i != '.':

ans += int(i) \* (3 \*\* d)

d -= 1

**if** i **in** bad:

f = **False**

**else**:

d = len(x) - 1

**for** i **in** x:

ans += int(i) \* (3 \*\* d)

d -= 1

**if** i **in** bad:

f = **False**

**if** **not** f:

ans = 'Число не в троичной системе счисления'

**except**:

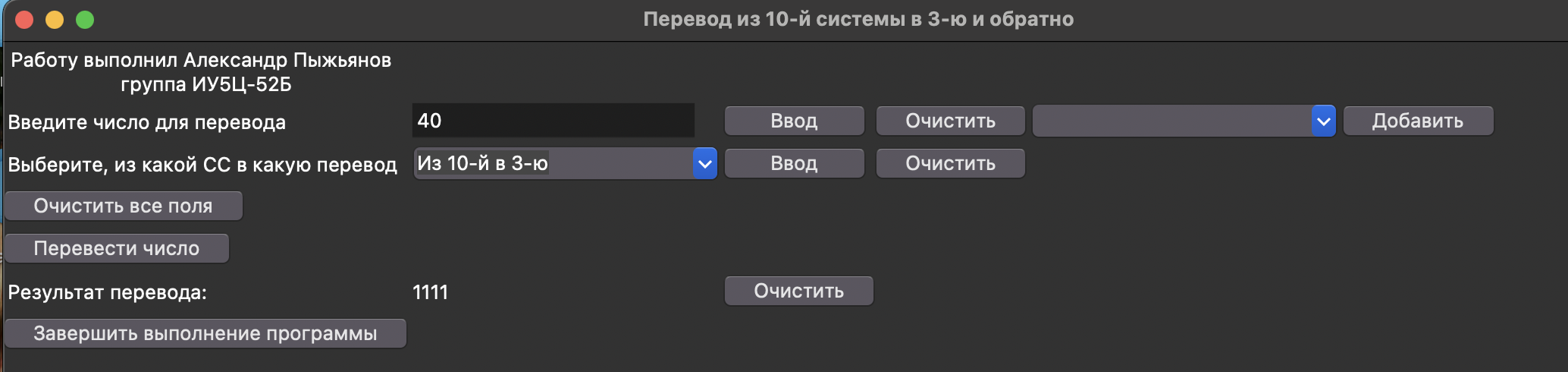
ans = 'Введено некорректное число'

**return** ans

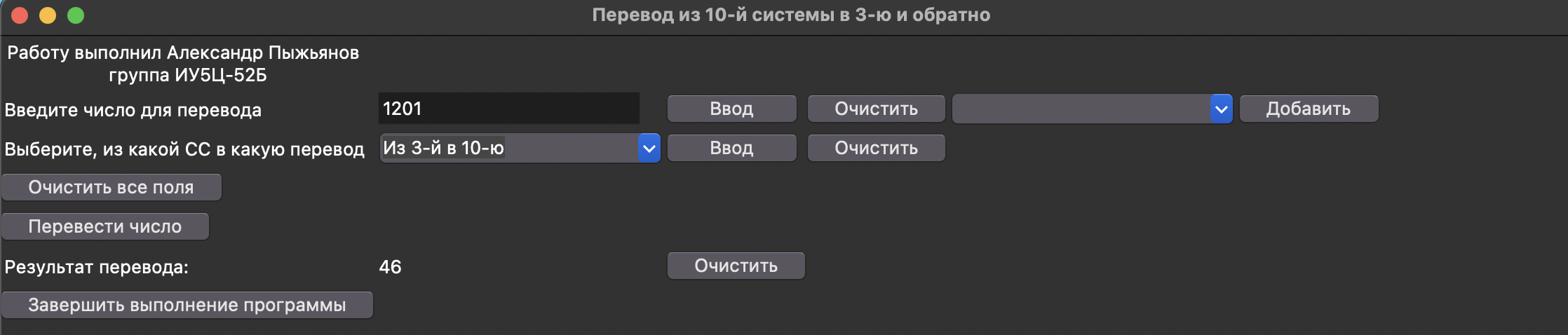
visual()

## Результат работы программы

Перевод из 10 в 3



Перевод из 3 в 10



Обработка исключений

