

Лабораторная работа №6

Разработка приложений с графическим пользовательским интерфейсом.

Порядок выполнения

1. Подготовка к выполнению работы

1. Создайте в своей рабочей папке папку с именем LR6. Все входные и выходные файлы, файлы с кодами программ сохраняйте в этой папке.
2. Создайте в папке LR6 текстовый документ «Отчет.doc».

2. Выполнение работы.

Напишите программы для решения задач согласно варианту, отладьте их, запустите, получите результаты. Поместите в отчет:

- тексты заданий,
- словесные описания алгоритмов,
- коды программ,
- скриншоты окон работающих приложений,

Задача 1. Первый список заполняется числами программно в соответствии с вариантом. По щелчку на кнопке надо перенести во второй список только числа, удовлетворяющие условию У1. Вывести количество чисел в первом и во втором списках

Вариант	Условие У1	Первый список	Второй список
1	Число представимо в виде $2^m 3^n$	18, 90, 31, 192, 65, 27, 143, 36, 21, 24, 212, 254, 48, 324 - 14 шт.	18 192 27 36 24 48 324 – 7 шт.
2	В числе есть одинаковые цифры, стоящие рядом.	1228, 303, 1000, 4854, 55, 1010, 28, 3117, 939, 1234, 8445, 777 – 12 шт.	1228, 1000, 55, 3117, 8445, 777 – 6 шт.
3	Число имеет ровно 3 собственных делителя (т.е. делителей, отличных от самого числа).	36, 144, 981, 1042, 1099, 146, 3291, 4703, 15, 5008, 8503, 640, 9638, 519 – 14 шт.	1042, 1099, 146, 3291, 15, 8503, 519 – 7 шт.
4	В числе есть одинаковые цифры, не обязательно стоящие	1238, 303, 1000, 4854, 56, 1010, 268, 3117, 939, 1234, 8445, 7707 –	303, 1000, 4854, 1010, 3117, 939, 8445, 7707 – 8 шт.

	рядом.	12 шт.	
5	Число является простым.	110, 41, 433, 168, 97, 1000, 101, 13, 5612, 369, 173, 450, 267, 31 – 14 шт.	41, 433, 97, 101, 13, 173, 31 – 7 шт.
6	В двоичной записи числа имеется не менее двух нулей.	193, 56, 79, 255, 98, 343, 51, 10, 286, 415, 31, 33, 499 – 13 шт.	193, 56, 98, 343, 286, 33 – 6 шт.
7	Число является составным.	110, 41, 433, 168, 97, 1000, 101, 13, 5612, 369, 173, 450, 267, 31 – 14 шт.	110, 168, 1000, 5612, 369, 450, 267 – 7 шт.
8	Число четырехзначное и его цифры расположены следующим образом: четная-нечетная-четная-нечетная.	1121, 2349, 6107, 3054, 8186, 4563, 1010, 2103, 8888, 4445, 8989, 2487, 3334, 6163, 2023 – 15 шт.	2349, 6107, 4563, 2103, 8989, 6163 – 6 шт.
9	Число имеет ровно 2 четных делителя, не совпадающих с самим числом.	624, 338, 16, 388, 18, 124, 8, 50, 98, 22, 578, 900, 722 – 13 шт.	338, 18, 8, 50, 98, 578, 722 - 7 шт.
10	Среди восьмеричных цифр числа есть хотя бы одна нечетная.	119, 260, 168, 1492, 134, 1024, 968, 1056, 1158, 566, 1123 – 11 шт.	119, 168, 1492, 968, 566, 1123 – 6 шт.
11	Число является степенью двойки.	23, 1022, 16, 192, 64, 2, 486, 18, 31, 128, 510, 512, 4196, 94, 4096 – 15 шт.	16, 64, 2, 128, 512, 4096 – 6 шт.
12	Число является палиндромом (читается одинаково слева направо и справа налево).	191, 56, 77, 255, 98, 343, 51, 10, 202, 41514, 31, 3773, 499, 6 – 14 шт.	191, 77, 343, 202, 41514, 3773, 6 – 7 шт.
13	Число является совершенным (равно	119, 496, 168, 28, 134, 1024, 8128, 1056, 1158,	496, 28, 6, 8128 – 4 шт.

	сумме своих собственных делителей)	6, 1123 – 11 шт.	
14	Число является «счастливым» (сумма цифр первой половины числа равна сумме цифр второй половины числа)	8228, 35, 1000, 4857, 55, 1030, 28, 3417, 939, 1634, 8879, 777 – 12 шт.	8228, 4857, 55, 1634, 8879 – 5 шт.
15	Количество всех делителей числа нечетно	18, 121, 75, 1369, 4, 570, 131, 1025, 1156, 442, 441, 1368, 289	121, 1369, 4, 1156, 441, 289 – 6 шт.
16	Цифры числа образуют возрастающую последовательность.	1467, 225, 874, 1269, 28, 11, 5690, 34567, 892, 13579, 555, 9145, 347 -13 шт.	1467, 1269, 28, 34567, 13579, 347 – 6 шт.

```
from tkinter import *
```

```
def p2m3n(x):
```

```
    while x % 2 == 0:
```

```
        x //= 2
```

```
    while x % 3 == 0:
```

```
        x //= 3
```

```
    return x== 1
```

```
def perenos():
```

```
    k = 0
```

```
    list2.delete(0, list2.size())
```

```
    for i in numbers:
```

```
        if p2m3n(i):
```

```
            list2.insert(END, i)
```

```
            k += 1
```

```
    lb2 = Label(text = 'Во втором списке ' + str(k))
```

```
    lb2.pack()
```

```
root = Tk()
```

```
root.geometry('400x250')
```

```
numbers
```

```
[18, 90, 31, 192, 65, 27, 143, 36, 21, 24, 212, 254, 48, 324]
```

```
list1 = Listbox()
```

```
list1.pack(side=LEFT)
```

=

```

for i in numbers:
    list1.insert(END,i)
list2 = Listbox()
list2.pack(side=RIGHT)
btn = Button(text = 'Перенос',command = perenos)
btn.pack(side=BOTTOM)
lbl = Label(root,text = 'в первом списке
'+str(list1.size()))
lbl.pack()

```

Задача 2. Первый список заполняется строками в соответствие с вариантом. При выделении строки в списке выводится эта строка, измененная согласно правилу П1. Если строка не отвечает правилу П1, то вывести соответствующее сообщение. Вывести количество строк в списке, отвечающих правилу П1.

Вариант	Исходный список строк	Правило П1
1	БПЛА могут, обладать разной, степенью автономности, от управляемых дистанционно, до полностью автоматических	Вставить перед каждой буквой 'С' и 'с' символ '+'
2	БПЛА могут решать, разведывательные задачи, (на сегодня это, основное их предназначение)	Перед каждой буквой 'А' и 'а' вставить пробел
3	Основным преимуществом БПЛА, является существенно меньшая, стоимость их создания, и эксплуатации	Вставить между всеми рядом стоящими буквами 'с' и 'т' символ '^'
4	Важным фактором, является то, что оператор боевого БПЛА, не рискует своей жизнью, в отличие от пилота, боевого самолёта	Удвоить каждую букву 'е' и 'о' в строке
5	Существенно меньшая, стоимость их создания, и эксплуатации, (при условии равной эффективности, выполнения поставленных задач)	Во всех цепочках из двух одинаковых букв вставить между этими буквами букву 'О'
6	Недостатком БПЛА, является уязвимость систем, дистанционного управления, что особенно важно, для БПЛА, военного назначения	Перед каждой цепочкой 'но', 'на', 'не' и 'ни' вставить точку
7	Управление БПЛА, может осуществляться, эпизодической подачей команд, или непрерывно (для ДПЛА)	После каждой буквы 'Л' и 'л' вставить символ '#'

8	Беспилотный летательный аппарат, также беспилотник; дрон ¹¹ —, летательный аппарат, без экипажа на борту	Заменить в строке буквы 'e' буквами 'E'
9	БПЛА верхнего, диапазона сложности, стоят от 5–6 млн. долл., тогда как стоимость, пилотируемого F-35, составляет около \$100 млн. долл.	Удвоить каждую цифру в строке
10	В качестве бортовой, аппаратуры управления, используются РС/104., Программное обеспечение пишется на языках, Си Си++, Модула-2, или Ада95	Заменить все цифры в строке нулями
11	Группа – I, Масса – 0-9 кг, Рабочая высота < 360 м, Скорость – 45-50 узлов, Пример – RQ-11 Raven	Вставить перед каждой цифрой знак '-'
12	В качестве бортовой, аппаратуры управления, используются РС/104., Программное обеспечение пишется на языках, Си; Си++; Модула-2 или Ада95	Заключить каждый знак в круглые скобки
13	Группа – II, Масса – 9-25 кг, Рабочая высота < 1050 м, Скорость – <250 узлов, Пример – ScanEagle	Заменить все цифры в строке символами '%'
14	Беспилотный летательный аппарат, можно выполнить, сколь угодно малых размеров, в то время как пилотируемый, невозможно сделать, меньше габаритов человека.	Заключить все буквы 't' в угловые скобки '<>'
15	БПЛА верхнего, диапазона сложности, стоят от 5–6 млн. долл., тогда как стоимость, пилотируемого F-35, составляет около \$100 млн. долл.	Вставить после каждой буквы 'a' символ '_'
16	Управление БПЛА, может осуществляться, эпизодической подачей команд, или непрерывно (для ДПЛА)	Заменить в строке буквы 'p' символами '!'

```
def modify():
    # получаем кортеж индексов выделенных строк
    selection = lst.curselection()
    # получаем строку для изменения
    s = lst.get(selection[0])
```

```

# меняем
s1 = ''
f = 0
for c in s:
    if c == 'C' or c == 'c':
        s1 += '+' + c
        f += 1
    else:
        s1 += c
# ВЫВОДИМ на метку
if f > 0:
    lbl.configure(text = s1)
else:
    lbl.configure(text = 'Нет букв "с"')
root = Tk()
root.geometry('600x300')
lst = Listbox(width= 30, font = ('Times New Roman',
15))
lst.pack(side = LEFT)

sp = ['БПЛА могут', 'обладать разной', 'степенью
автономности', \
      'от управляемых дистанционно', 'до полностью
автоматических']
for i in sp:
    lst.insert(END,i)
lbl = Label(text = '----',font = ('Times New Roman',
15))
lbl.pack(side = TOP)
btn = Button(text='Пуск', font = ('Times New Roman',
15),command= modify)
btn.pack(side = БОТТОМ)
root.mainloop()

```