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

**Отчет по лабораторной работе №3 по курсу**

**«Парадигмы и конструкции языков программирования»**

#### Тема работы: " Исследование

#### методов сортировки"

|  |  |
| --- | --- |
| 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 сортировкой массива.

## Практическое задание

Написать программу для демонстрации работы метода вставок с бинарным поиском сортировки на примере массива целых чисел.

Программа должна состоять из двух частей (этапов работы) и выполнять два действия **последовательно**:

1. сначала отсортировать заданный пользователем массив для доказательства

корректности работы алгоритма;

2. затем составить таблицу замеров времени сортировки списков трёх различных (заданных пользователем) размерностей и количества перестановок в каждом из них.

В части 2 для каждой размерности списка необходимо исследовать:

● случайный список,

● отсортированный список,

● список, отсортированный в обратном порядке.

В результате должна получиться поясняющая таблица.

1. Текст программы должен сопровождаться необходимыми комментариями,

поясняющими основные действия и назначение переменных.

2. Программа должна выдавать корректные данные для любых допустимых

входных данных (при этом гарантируется, что на вход подаются только

числовые значения).

3. При выводе числовых значений отображать 5-7 значащих цифр числа.

4. При вводе данных должно выводиться приглашение, при выводе – пояснение, краткие и однозначно интерпретируемые пользователем.

5. Приглашение и пояснения должны формулироваться с заглавной буквы и обычно заканчиваются двоеточием и пробелом.

6. Исходный код должен быть оформлен согласно стандарту PEP 8

## Листинг программы

**import** random

**import** time

**def** insertonsortwithbinnsearh(arr):

k = 0

**for** i **in** range(1, len(arr)):

cur = arr[i]

lo = 0

hi = i

**if** lo == hi:

lo += 1

**else**:

**while** lo < hi:

mid = (lo + hi) // 2

**if** cur < arr[mid]:

hi = mid

**else**:

lo = mid + 1

j = i

**while** j > lo **and** j > 0:

arr[j] = arr[j - 1]

j -= 1

k += 1

k += 1

arr[lo] = cur

**return** k

**def** zamer(n):

a = []

**for** i **in** range(n):

a.append(i + 1)

t1 = time.time()

k1 = insertonsortwithbinnsearh(a)

t1 = time.time() - t1

a = []

**for** i **in** range(n):

a.append(random.randint(1, n \* 10))

t2 = time.time()

k2 = insertonsortwithbinnsearh(a)

t2 = time.time() - t2

a = []

**for** i **in** range(n):

a.append(i + 1)

a = a[::-1]

t3 = time.time()

k3 = insertonsortwithbinnsearh(a)

t3 = time.time() - t3

**return** int(t1 \* 1000), k1, int(t2 \* 1000), k2, int(t3 \* 1000), k3

a = []

**try**:

n = int(input())

**if** n > 0:

**for** i **in** range(n):

a.append(random.randint(1, 100))

print(\*a)

insertonsortwithbinnsearh(a)

print(\*a)

**else**:

print("n отрицательное")

**except**:

print("Введено неверное значение")

**try**:

n1 = int(input())

n2 = int(input())

n3 = int(input())

t1, k1, t4, k4, t7, k7 = zamer(n1)

t2, k2, t5, k5, t8, k8 = zamer(n2)

t3, k3, t6, k6, t9, k9 = zamer(n3)

print(" " \* 13, "{n1:^15} {n2:^15} {n3:^15}".format(n1 = n1, n2 = n2, n3 = n3))

print(" " \* 13, "время перестановки время перестановки время перестановки")

print("Упорядоченный", "{t1:^5}{k1:^14}{t2:^5}{k2:^14}{t3:^5}{k3:^14}".format(t1 = t1, k1 = k1, t2 = t2, k2 = k2, t3 = t3, k3 = k3))

print("Список")

print("Случайный ", "{t4:^5}{k4:^14}{t5:^5}{k5:^14}{t6:^5}{k6:^14}".format(t4 = t4, k4 = k4, t5 = t5, k5 = k5, t6 = t6, k6 = k6))

print("Список")

print("Упорядоченный", "{t7:^5}{k7:^14}{t8:^5}{k8:^14}{t9:^5}{k9:^14}".format(t7 = t7, k7 = k7, t8 = t8, k8 = k8, t9 = t9, k9 = k9))

print("в обратном")

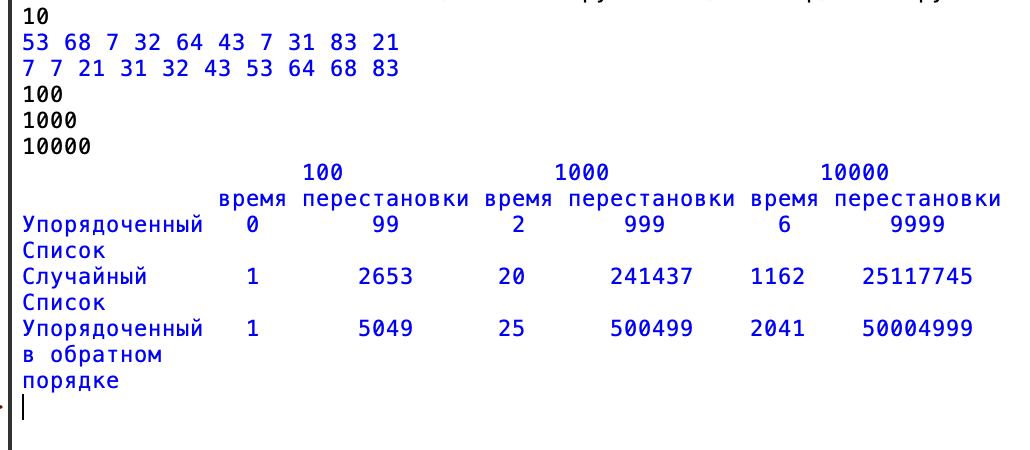
print("порядке")

**except**:

print("Одна или несколько размерностей введены неверно")

## Результат работы программы

Рабочий пример



Обработка исключений

