

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №7

СОРТИРОВКИ МАССИВОВ

1. Цель и порядок работы

Цель работы - получение практических навыков по использованию алгоритмов сортировки одномерных массивов (списков).

Порядок выполнения работы:

- ознакомиться с описанием практического занятия;
- написать программы, согласно вариантам;
- продемонстрировать работу преподавателю.

2. Теоретический материал

Кроме материалов лекции, можно воспользоваться материалами по данной теме из следующих источников:

1. <https://timeweb.com/ru/community/articles/vidy-algoritmov-sortirovki-v-python>
2. <https://senior.ua/articles/8-metodov-sortirovok-python-kotorye-obязatelno-nuzhno-znat>
3. <https://pythonru.com/osnovy/top-5-algoritmov-sortirovki-na-python>
4. <https://pythonist.ru/algoritmy-sortirovki-s-python/>
5. <https://tproger.ru/translations/sorting-algorithms-in-python/>
<https://pythonworld.ru/osnovy/cikly-for-i-while-operatory-break-i-continue-volshebnoe-slovo-else.html>

3. Контрольные вопросы

1. Что такое алгоритм сортировки?
2. Объясните суть сортировки массива методом «пузырька».
3. Объясните суть сортировки массива методом выбора.
4. Объясните суть сортировки массива методом вставки
5. Какое максимальное число перестановок может быть в сортировке массива методом «пузырька».
6. Какое максимальное число перестановок может быть в сортировке массива методом выбора.
7. Какое максимальное число перестановок может быть в сортировке массива методом вставки
8. Какое максимальное число сравнений может быть в сортировке массива методом «пузырька».
9. Какое максимальное число сравнений может быть в сортировке массива методом выбора.
10. Какое максимальное число сравнений может быть в сортировке массива методом вставки

4 Задание

1. Изучите теоретический материал.
2. Ответьте на теоретические вопросы.
3. Напишите программы в соответствии с заданиями из пункта 5 на языке программирования Python.
4. Проверьте работоспособность программ.
5. Отладьте и протестируйте программы.
6. Сделанную работу сдайте преподавателю. При выставлении баллов за работу учитывается: правильность написания кода, полнота и правильность ответов учащегося на вопросы преподавателя.

5 Задания к лабораторной работе

5.1 Задание для работы в аудитории

Указания к заданию:

- рассмотрите теоретический материал по алгоритмам сортировки;
- используя алгоритмы сортировки напишите программу сортировки массива из n элементов (n вводится с клавиатуры) следующим образом: отсортировать массив по не убыванию с помощью алгоритма сортировки «пузырек» до элемента массива, индекс которого вводится с клавиатуры (необходимо осуществить проверку введенного индекса), оставшуюся часть массива отсортировать по не возрастанию с помощью алгоритма сортировки массива выбором;
- заполнение массива осуществить случайным образом.

5.2 Домашнее задание

Указания к домашнему заданию:

1. Напишите программу, которая выполняет следующие функции:
 - - заполнение элементов массива вещественными числами с заданной размерностью случайным образом;
 - - сортировку массива каждым из 3 способов (пузырьковая сортировка, сортировка выбором, сортировка вставкой);
 - - подсчет времени сортировки и количества перестановок.
2. Провести эксперимент сортировки массива со следующим количеством элементов: 100, 1000 и 10000. Для проведения эксперимента необходимо произвести по 5 запусков каждого алгоритма и выбрать наилучшее время. Выбранный наилучший результат занести в таблицу. Сортировку осуществлять с одним и тем же набором данных.

Вид сортировки	100		1000		10000	
	время	кол-во перестановок	время	кол-во перестановок	время	кол-во перестановок
Bubble sor						
Selection sort						
Insertion sort						

3. Время выполнения сортировки можно определить следующим образом:

```
from datetime import datetime
start_time = datetime.now()
# текст программы
end_time = datetime.now()
print('Время выполнения: {}'.format(end_time - start_time))
```

4. Оформите отчет об эксперименте следующим образом:

Постановка задачи

Привести постановку задачи, определить входные и выходные параметры.

Аппаратная часть эксперимента

Привести описание компьютера на котором проводился эксперимент (вид процессора, тактовая частота, количество ядер, количество оперативной памяти и т.д.).

Текст программы на языке Python

Привести полный текст программы с комментариями поясняющие ее работу.

Результаты эксперимента

Привести результаты эксперимента в виде таблицы, вывод по каждой размерности массива с указанием лучшей сортировки и пояснениями.