МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**ЗВІТ**

**з лабораторної роботи № 2**

**«Алгоритми сортування**

**121 – Інженерія програмного забезпечення. 2021–2022 н.р.»**

# Дисципліна «Алгоритми та структури даних 1. Основи алгоритмізації»

**Виконала :**

студентка 1-го курсу,

групи КП-11,

спеціальності 121 – Інженерія

програмного забезпечення

*Кирильчук Олександра Артурівна*

Київ – 2021

**Контрольні запитання**

**1.Що таке міра відсортованості?**

Міра відсортованості – це міра, яка визначає наскільки даний масив вже є відсортованим.

**2. Що таке стійкість алгоритму?**

Стійкість – здатність не виконувати обмін однакових елементів.

**3. Чому алгоритм сортування бульбашкою є доволі повільним?**

Позиція елементів, що підлягають сортуванню відіграє велику роль у питанні продуктивності даного алгоритму. Великі елементи на початку списку не викликають проблеми, оскільки вони досить швидко переміщуються на свої місця. Однак, малі елементи у кінці списку переміщуються на його початок дуже повільно. Це призвело до того, що ці типи елементів було названо **кроликами і черепахами**, відповідно.

Для алгоритму сортування бульбашкою вірним буде запис:

**О(n2)** – у *найгіршому* випадку, коли вхідні дані відсортовані у зворотньому порядку, обчислювальна складність алгоритму буде квадратично залежати від кількості елементів.

**Θ(n2)** – у *середньому* випадку, коли дані перемішані у випадковому порядку, обчислювальна складність алгоритму також буде квадратично залежати від кількості елементів.

**Ω(n)** – у *найкращому* випадку, коли вхідні дані вже відсортовані, нам все одно доведеться здійснити один прохід масивом, тобто виконати n операцій.

**4. Які є модифікації алгоритму сортування бульбашкою?**

**Сортування змішуванням** – один із різновидів алгоритму сортування бульбашкою. Відрізняється від сортування бульбашкою тим, що сортування відбувається в обох напрямках, міняючи напрямок при кожному проході. Даний алгоритм лише трохи складніший за сортування бульбашкою, однак, вирішує так звану проблему "черепах".

**Шейкерне сортування - двонаправлене бульбашкове сортування. У цьому випадку алгоритм обробляє масив спочатку зліва направо, переміщуючи найбільший елемент у кінець масиву, а потім справа наліво, переміщуючи найменший елемент на початок масиву.**

**Сортування гребінцем** - це покращений варіант сортування бульбашкою, сортування нестійке. Відмінність полягає в тому, що порівнюються не два сусідні елементи, а два елементи розташовані на деякій відстані один від одного. Такий підхід дозволяє на початку сортування перемістити елементи з меншими значеннями з кінця масиву на його початок, за рахунок чого відбувається зменшення обчислювальної складності. Відстань між елементами, що порівнюються, обирається не випадково, а з врахуванням фактора зменшення.

**Непарно-парне сортування**, або «цеглинне» сортування (Odd-even sort або Brick sort) – ще одна модифікація сортування бульбашкою. Ідея алгоритму полягає у поділі алгоритму бульбашкою на дві фази: непарну та парну. У непарній фазі відбувається сортування відносно елементів, які мають непарні індекси. У парній фазі сортування виконується відносно елементів, які мають парні індекси.

**5. Які є модифікації алгоритму сортування вставкою?**

**Сортування Шелла** (Shell sort) – вдосконалення сортування вставками, яке є аналогічним до сортування гребінцем. Ідея алгоритму полягає у тому, що спочатку сортуються елементи, які знаходяться на максимально можливій відстані один від одного, після чого ця відстань поступово скорочується. Значення відстані між елементами по черзі обирається з певної послідовності значень, яка впливає на ефективність сортування.

<https://habrastorage.org/webt/ig/rb/aa/igrbaajcuoxuj4q-l-38x0qsoq4.gif>

**Па́рная сортировка простыми вставками**  
Модификация простых вставок, разработанная в тайных лабораториях корпорации Oracle. Эта сортировка входит в пакет JDK, является составной частью Dual-Pivot Quicksort. Используется для сортировки малых массивов (до 47 элементов) и сортировки небольших участков крупных массивов.  
В буфер отправляются не один, а сразу два рядом стоя́щих элемента. Сначала вставляется больший элемент из пары и сразу после него метод простой вставки применяется к меньшему элементу из пары.

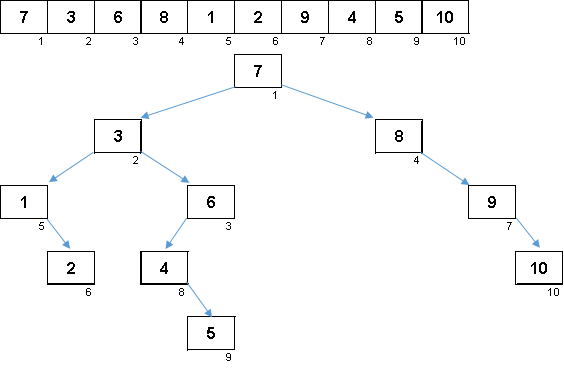
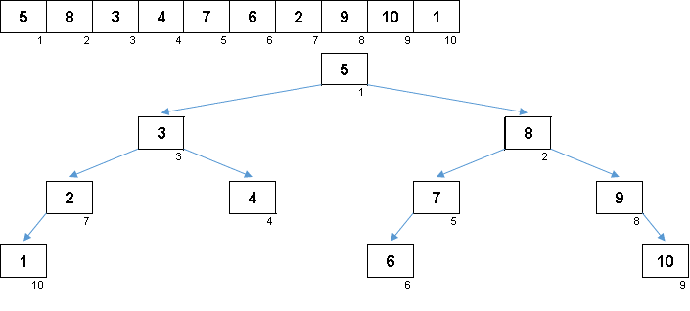
<https://habrastorage.org/webt/fi/iv/q6/fiivq621vdspxj9bxxq0pqe3jne.gif>

**Сортування деревом**  
Сортування деревом зарахунок додаткової пам'яті швидко вирішує питання з додаванням чергового елемента у відсортовану частину масиву. Причому в ролі відсортованої частини масиву виступає бінарне дерево. Дерево формується буквально на льоту під час перебору елементів.

Елемент порівнюється спочатку з коренем, а потім і з більш вкладеними вузлами за принципом: якщо елемент менше ніж вузол - то спускаємося лівою гілкою, якщо не менше - то правою. Побудоване за таким правилом дерево можна легко обійти так, щоб рухатися від вузлів з меншими значеннями до вузлів з великими значеннями (і таким чином отримати всі елементи в зростаючому порядку).

<https://habrastorage.org/webt/uo/6e/d8/uo6ed8lcpyvdpg9fzfwsye2pnuu.gif>

Основна заковика сортувань вставками(витрати на вставку елементу на своє місце у відсортованій частині масиву) тут вирішена, побудова відбувається цілком оперативно. В усякому разі для звільнення точки вставки не треба повільно пересувати каравани елементів як в попередніх алгоритмах.+. Коли виходить красива симетрична ялиночка(так зване ідеально збалансоване дерево) як в анімації, то вставка відбувається швидко, оскільки дерево в цьому випадку має мінімально можливу вкладеність рівнів. Але збалансована(чи хоч би близька до такої) структура з рандомного масиву виходить рідко. І дерево, швидше за все, буде неідеальне і незбалансоване - з перекосами, заваленим горизонтом і надмірною кількістю рівнів.

*Рандомний масив зі значеннями від 1 до 10. Елементи в такому порядку генерують незбалансоване двійкове дерево:*  
  
  
  
Дерево мало побудувати, його ще треба обійти. Чим більше незбалансованості — тим більше буксуватиме алгоритм з обходу дерева. Тут як скажуть зірки, рандомний масив може породити як потворну корч (що більш ймовірно) так і деревоподібний фрактал.  
  
*Значення елементів ті самі, але порядок інший. Генерується збалансоване двійкове дерево:*  
  
 Проблему незбалансованих дерев вирішує сортування вивертанням, яке використовує особливий різновид бінарного дерева пошуку - **splay tree**. Це чудове дерево-трансформер, яке після кожної операції перебудовується на збалансований стан.