Сортування вставкою (СІ)

Вхідна послідовність ділиться на дві частини : посортована, не посортована  
Спочатку посортована частина містить перший елемент вхідної послідовності. Загалом СІ (N-1) Іі, і=1,2,..N-1

На кожному з етапів розмірність зменшується на одиницю. Це відбувається шляхом вставки першого елемента не посортованої частини послідовності, на відповідну позицію посортованої частини. 33

Схема етапу І3 X[4]

S0

S1

S2 посортована частина

не посортована частина

До посортованої під послідовності S' [і], і=0,1,2,3,4 вставляється елемент Х[4] На виході отримаємо підпослідовність S[i], i=1,2,3 яка буде посортованою частиною для елементів X.  
  
Процес завершується тільки після вставки останнього елемента.  
Аглоритм сортування вставкою N=7 (мал. див в Івана)

------------------

Алгоритми сортування вставками і сортування перестановками маю подібну стрруктуру. Вони складаються з однокової кількості етапів які є структурно ідентичні але виконуються в зворотній послідовності. Характерна особливість СІ це зменшення кількості порівнянь, порівняно з алгоритмом СП. При вставці кожного елемента частина не посортованої до посортованої, виконується тільки одна базова операція, кожна наступна базова операція виконується тільки якщо в попередній відбулася перестановка.   
 Відсутність перестановки в біжучій базові операцій символізує про закінчення процесу вставки.  
  
Найменша кількість порівнянь (оптимістична складність) в алгоритмі СІ N-1, а найбільша к-сть порівнянь (песимістична складність)   
Ксі (N)=1+2...+(N-1)= ]N:2 [ (N-1)

Порівняння ефективності елементарних алгоритмів сортування

На перший погляд, найефективніший є алгоритм сортування вставкою бо він в песимістичному випадку = сортуванням перестановкою і вибору. Але у випадках даних зі складною структурою, зі складною операцією перестановки , ефективнішими можуть бути сортування вибору, займає тільки одної сортування перестановки. Найменш ефективний для ітераційних процесорів алгоритм сортування перестановкою, має перевагу над алгоритмом сортування вибором, при реалізації на потокових процесорах.

Фактично ефективність кожного алгоритму можна оцінити тільки експериментально, для кожної архітектури обчислювача.

Завдання:

1. Продемонструвати роботу алгоритму сортування перестановкою, вибором, вставкою на прикладі числової послідовності.

2. Запропонувати способи покращення алгоритмів сортування перестановки, вибору, вставки.

3. Обґрунтувати оптимальність алгоритмів MIN1 і MAX1.

4. За аналогію алгоритму MIN\_MAX2M побудувати алгоритм одночасного вибору двох найменших або двох найбільших елементів послідовності.

5. Дослідити для якої послідовності, алгоритм Si вимагає найменшої кількості порівнянь , а для якої найбільшої.

6. Написати код алгоритмів сортування, перестановки і вибору.

Напівшвидкі алгорим сортування