

Лабораторна робота №13-14

Сортування одновимірного масиву

Мета: набуття практичних навиків роботи з одновимірними масивами, а саме сортування елементів масиву різними методами. Здійснення порівняння та аналізу ефективності використовуваних методів сортування.

Література

Войтенко В. В., Морозов А. В. С\С++ Практика програмування. Навчально-методичний посібник - Житомир: ЖДТУ, 2003. – 324 с.

Зміст роботи

Завдання:

1. Згенеруйте одновимірний масив і виведіть його на екран.
2. Відсортуйте масив кожним описаним методом.
3. Порівняйте швидкість метода сортування при різних значеннях n (кількість елементів масиву) маленьких, середніх, великих. Доведіть розмір масиву до 100000 і більше елементів.
4. Запускаючи програму не менше десяти разів для кожного методу, отримати час t сортування. Побудувати залежність $t=f(n)$ на одному графіку для різних методів сортування (у табличному і графічному виглядах).
5. Які з вивчених методів сортування дають найкращий результат для великих масивів? Який метод найгірший?

Методичні рекомендації

Сортування масиву - один з найбільш розповсюджених процесів обробки даних. Завдяки йому здійснюється розміщення об'єктів у визначеному порядку, наприклад, чисел за зростанням або за спаданням їх значень, прізвищ у алфавітному порядку тощо. Існують різні методи сортування, серед яких – сортування обміном (метод «бульбашки», «шейкер-сортування»), сортування вибором, сортування вставками, швидке сортування, сортування Шелла, пірамідальне сортування, сортування обчисленням адреси, сортування порозрядним групуванням тощо. Ці методи відрізняються швидкістю отримання результату, складністю і універсальністю.

Сортування обміном

Метод «сортування обміном» ґрунтується на перестановці сусідніх елементів. Для впорядкування елементів масиву здійснюються повторні проходи по масиву. Переміщення елементів масиву здійснюється таким чином: масив переглядається зліва направо, здійснюється порівняння пари сусідніх елементів; якщо елементи в парі розміщені в порядку зростання, вони лишаються без змін, а якщо ні міняються місцями. В результаті 1-го проходу найбільше число буде поставлено в кінець масиву. У 2-му проході

операції виконуються над елементами з першого до (N-1)-ого, у 3-му – від першого до (N-2)-ого і т.п. Впорядкування масиву буде закінчено, якщо при проході масиву не виконається жодної перестановки елементів масиву. Факт перестановки фіксується за допомогою деякої змінної (fl), яка на початку має значення 0 і набуває значення 1, коли виконається перестановка в будь-якій парі.

Масив до сортування	20	5	67	3	4	-2	5	10
1	5	20	3	4	-2	5	10	67
2	5	3	4	-2	5	10	20	67
3	3	4	-2	5	5	10	20	67
4	3	-2	4	5	5	10	20	67
5	-2	3	4	5	5	10	20	67

```

const int n=8; int c, fl;
int array[]={20,5,67,3,4,-2,5,10};
do
{
    fl=0;
    for (int i=1;i<n;i++)
    if (array[i-1]>array[i])
    {
        c=array[i];
        array[i]=array[i-1];
        array[i-1]=c;
        fl=1;
    }
} while (fl);

```

Сортування методом вибору

Сутність методу: масив переглядається перший раз, знаходиться мінімальний елемент масиву і міняється місцями з першим елементом. Другий раз масив переглядається, починаючи з другого елемента. Знову знаходиться мінімальний елемент і міняється місцями з другим.

Даний процес виконується доти, поки не буде поставлений на місце N-1 елемент.

Масив до сортування	20	5	67	3	4	-2	5	10
1	-2	5	67	3	4	20	5	10
2		3	67	5	4	20	5	10
3			4	5	67	20	5	10
4				5	5	20	67	10
5					5	10	67	20
6						10	20	67

```

const int n=8; int imin, a;
int array[]={20,5,67,3,4,-2,5,10};
for (int i=0;i<n-1;i++)
{

```

```

    imin=i;
    for (int j=i+1;j<n;j++)
        if (array[j]<array[imin]) imin=j;
    a=array[i];
    array[i]=array[imin];
    array[imin]=a;
    printf("\n");
}

```

Сортування вставками

При використанні даного методу на *i*-му етапі відбувається «вставка» елемента *a[i]* в потрібну позицію серед елементів *a[1], ..., a[i-1]*, які вже впорядковані. Після цієї вставки перші *i* елементів будуть впорядковані.

Масив до сортування	20	5	67	3	4	-2	5	10
1	5	20						
2	5	20	67					
3	3	5	20	67				
4	3	4	5	20	67			
5	-2	3	4	5	20	67		
6	-2	3	4	5	5	20	67	10
7	-2	3	4	5	5	10	20	67

```

const int n=8; int c;
int array[]={20,5,67,3,4,-2,5,10};
for (int i=1;i<n;i++)
{
    c=array[i];
    for (int j=i-1;j>=0 && array[j]>c;j--)
    {
        array[j+1]=array[j];
        array[j]=c;
    }
}

```

Сортування методом Шелла

Головний недолік простих методів—обмін ведеться в основному між сусідніми елементами. Тому бажано робити якомога ширші обміни.

Метод Шелла—метод сортування включеннями з відстанями, що зменшуються.

На першому етапі масив уявно ділиться на підмасиви наприклад $size =$, $step = size / 2 = 4$, кожен з них впорядковується окремо.

На другому етапі підмасиви утворюються елементами через один.

Після цього весь масив сортується разом. За рахунок попередніх етапів він стає вже близьким до відсортованого, тому обмінів необхідно вже не так багато. Цей метод дає кращі результати, якщо розподіл на підмасиви роблять

кроками, що не є степенями двійки, а, навпаки, не є множниками один одного.

Масив до сортування	20	5	67	3	4	-2	5	10
1	4	-2	5	3	20	5	67	10
2	4	-2	5	3	20	5	67	10
3	-2	3	4	5	5	20	10	67

```
int step = size / 2;
while (step > 0)
{
for (int i = 0; i < (size - step); i++)
{
int j = i;
while (j >= 0 && a[j] > a[j + step])
{
int c = a[j];
a[j] = a[j + step];
a[j + step] = c;
j--;
}
}
step = step / 2;
}
```

Функція clock() для виміру часу виконання програми.

```
#include <time.h>
float dd;
clock_t start, finish;
start = clock(); // час початку виконання програми

// ... код програми...

finish = clock(); // час завершення виконання програми
dd = (float)(finish - start) / CLOCKS_PER_SEC;
// час виконання програми
```

Контрольні питання:

1. Які методи сортування ви знаєте? В чому їх сутність?
2. Поясніть алгоритм методу сортування *обміном*.
3. Наведіть приклад фрагмента коду, що здійснює сортування елементів масиву обміном у порядку зростання (починати сортування з першого елементу)
4. Поясніть алгоритм методу сортування *вставками*.
5. Наведіть приклад фрагмента коду, що здійснює сортування цілих чисел вставками у порядку зменшення (починати сортування з першого елементу)
6. Поясніть алгоритм методу сортування методом *вибору*.
7. Наведіть схему роботи сортування вибором (сортування починати з початку).

8. Наведіть приклад фрагмента коду, що здійснює сортування дійсних чисел вибором у порядку зменшення (починати сортування з кінця)
9. Наведіть схему роботи сортування методом обміну.
10. Як оцінюються методи сортування?
11. Наведіть алгоритм сортування методом Шелла. Які його переваги.

Завдання на самостійну роботу:

1. Опрацювати теоретичний матеріал (лекція 12).
2. Реалізувати методи сортування обміном, вибору, вставкам та Шелла для масивів дійсних типів даних за спаданням, та порівняти їх швидкодію.
3. *Дано масив із 16 чисел.
 - Розписати покроково результат пірамідального сортування і подати у вигляді таблиці.
 - Побудувати дерево.
4. *Реалізувати метод пірамідального сортування на мові програмування Сі.
5. Визначити яка схема належить якому методу сортування.

