

Лабораторна робота №6. Сортування злиттям

Мета: розробити програмну реалізацію алгоритму сортування злиттям. Провести аналіз алгоритму.

Теоретичні відомості

Сортування злиттям. Сортування злиттям (mergesort) є чи не найбільш популярним прикладом успішного застосування методу декомпозиції. Алгоритм сортує заданий масив $A[0..n-1]$ шляхом поділу його на дві половини, $A[0..\lfloor n/2 \rfloor - 1]$ і $A[\lfloor n/2 \rfloor .. n - 1]$, рекурсивного сортування кожної половини й злиття двох відсортованих половин в один відсортований масив.

Алгоритм Mergesort ($A[0..n-1]$)

// Рекурсивно сортує масив $A[0..n-1]$.

// Вхідні дані: масив $A[0..n-1]$ елементів, що мають бути впорядковані.

// Вихідні дані: відсортований масив $A[0..n-1]$ у неспадному порядку.

```
if  $n > 1$ 
```

```
    Копіювати  $A[0..\lfloor n/2 \rfloor - 1]$  в  $B[0..\lfloor n/2 \rfloor - 1]$ 
```

```
    Копіювати  $A[\lfloor n/2 \rfloor .. n - 1]$  в  $C[0..\lfloor n/2 \rfloor - 1]$ 
```

```
    Mergesort( $B[0..\lfloor n/2 \rfloor - 1]$ )
```

```
    Mergesort( $C[0..\lfloor n/2 \rfloor - 1]$ )
```

```
    Merge( $B, C, A$ )
```

Злиття двох відсортованих масивів можна виконати в такий спосіб. Два покажчики (індекси масивів) після ініціалізації вказують на перші елементи масивів, що зливаються. Потім елементи, на які вказують покажчики, порівнюються, і менший з них додається в новий масив. Після цього індекс меншого елемента збільшується, і він вказує на елемент, що безпосередньо слідує за тільки що скопійованим. Ця операція повторюється доти, поки не буде вичерпаний один з масивів, що зливаються, після чого елементи, що залишилися, другого масиву просто додаються в кінець нового масиву.

Алгоритм Merge ($B[0..p-1]$, $C[0..q-1]$, $A[0..p+q-1]$)

// Виконується злиття двох відсортованих масивів в один.

// Вхідні дані: відсортовані масиви $B[0..p-1]$ і $C[0..q-1]$.

// Вихідні дані: відсортований масив $A[0..p+q-1]$, що складається з елементів масивів B і C .

```
i=0, j=0, k=0
```

```
while  $i < p$  and  $j < q$  do
```

```
    if  $B[i] \leq C[j]$ 
```

```
         $A[k] = B[i]$ 
```

```

        i++
    else
        A[k]=C[j]
        j++
    k++
if i=p
    Копіювати C[j..q-1] в A[k..p+q-1]
else
    Копіювати B[i..p-1] в A[k..p+q-1]

```

Завдання до лабораторної роботи

Для виконання поставленої мети, необхідно вирішити наступні задачі:

1. Реалізувати алгоритм *Mergesort*.
2. Провести математичний аналіз алгоритму.
3. Провести емпіричний аналіз алгоритму.
4. Реалізувати інтерфейс користувача. Функціональні можливості інтерфейсу:
 - виводити запрошення на введення об'єму випадкової послідовності. Виводити генеровану випадкову послідовність у файл.
 - виводити відсортовану послідовність у файл із тим же ім'ям, але іншим розширенням.

Контрольні запитання

1. Що таке метод розробки (проекування) алгоритмів?
2. У чому полягає особливість методу декомпозиції?
3. Оцінка часу роботи алгоритму в найкращому разі, середньому, гіршому.
4. Недоліки методу сортування злиттям.