

Будь ласка, зверніть увагу! Це завдання на оцінку, яка буде враховуватися для отримання сертифікату.

Для виконання кожного практичного завдання у вас є тільки 5 спроб! Зарахована буде оцінка за останню спробу.

Постановка задачі

В цьому завданні вам необхідно реалізувати декілька модифікацій алгоритму швидкого сортування. Для перевірки коректності роботи вашої реалізації алгоритму буде використовуватись кількість порівнянь елементів під час роботи алгоритму. Так, для даного алгоритму порівняння елементів відбувається в процедурі Partition, коли елементи порівнюються з опорним елементом (рядок 4).

```

Partition(A, p, r)
1  x ← A[r]
2  i ← p - 1
3  for j ← p to r-1
4      do if A[j] ≤ x
5          then i ← i + 1
6              Обміняти A[i] ↔ A[j]
7  Обміняти A[i+1] ↔ A[r]
8  return i + 1
    
```

Зверніть увагу, що лічильник кількості порівнянь елементів можна збільшувати не безпосередньо перед рядком №4, а один раз за виклик процедури, при цьому збільшивши лічильник на величину $r - p$.

Увага! Процедура розбиття може бути реалізована різними способами, тому в цьому завданні ми просимо вас реалізувати її саме так, як наведено у лістингу вище, аби кількість порівнянь була однаковою з еталоном.

ВХІДНІ ТА ВИХІДНІ ФАЙЛИ

В цьому завданні на вхід вашої програми буде подаватись файл з 10 000 елементів. Його формат наступний: першим елементом є кількість N елементів масиву; наступні N рядків файлу відповідають елементам вхідного масиву.

Ви повинні будете реалізувати три модифікації алгоритму швидкого сортування. Для самостійної перевірки вам надається [три приклади вхідного масиву](#) (відповідні файли мають такий самий формат, як описано вище). Для кожного вхідного файлу є вихідний файл, який містить три числа, які відповідають кількостям порівнянь під час реалізації всіх трьох модифікацій алгоритму, що описані нижче.

ЗАВДАННЯ №1 (1/1 бал)

В першому завданні ви повинні запрограмувати алгоритм швидкого сортування таким чином, щоб опорним елементом в процедурі Partition завжди обирався останній елемент поточного масиву. Вам потрібно буде підрахувати загальну кількість порівнянь елементів в процедурі розбиття під час роботи алгоритму над масивом, який заданий у вхідному файлі.

Введіть загальну кількість порівнянь елементів в процедурі Partition:

150262

Перевірка

Зберегти

Показати відповідь

Ви використали 2 з 5 можливостей надіслати свої матеріали на розгляд.

ЗАВДАННЯ №2 (1/1 бал)

Це завдання відрізняється від першого тим, що в якості опорного елементу в процедурі Partition завжди обирається перший елемент поточного масиву. Для цього достатньо поміняти місцями перший та останній елементи та викликати основну версію процедури Partition, яка працює з останнім елементом в якості опорного. (Примітка: якщо ви спробуєте по-іншому реалізувати дане завдання, то можете отримати інші значення кількості порівнянь елементів.)

Вхідні дані для цього завдання такі самі, що й для завдання №1.

Введіть загальну кількість порівнянь елементів в процедурі Partition:

159864

159864

Перевірка

Зберегти

Показати відповідь

Ви використали 2 з 5 можливостей надіслати свої матеріали на розгляд.

ЗАВДАННЯ №3 (1/1 бал)

В останньому завданні вам потрібно реалізувати модифікацію алгоритму швидкого сортування, коли в якості опорного обирається медіана серед трьох елементів поточного масиву: першого, останнього та середнього. В термінах лістингу процедури Partition, який наведений вище, це будуть елементи за індексами p , r та $\lfloor \frac{p+r}{2} \rfloor$. Медіаною є середній за значенням елемент серед множини елементів.

Наприклад, якщо масив містить елементи 8 2 4 5 7 1, то необхідно розглянути перший (8), останній (1) та середній (4) елемент; медіаною в цьому випадку буде елемент 4, який має індекс 3 у вхідному масиві.

Щоб отримати такі самі результати, що й контрольні значення, вам необхідно знайти вказану медіану, поміняти її з останнім елементом в масиві та викликати стандартну процедуру Partition, яка працює з останнім елементом в якості опорного.

Якщо ви уважно прочитали умову, то у вас виникло питання, чи потрібно враховувати у загальне порівняння елементів ті порівняння, які відбуваються при визначенні медіани. Відповідь на це питання: Ні, не потрібно.

Увага! Також може постати питання, що робити, коли досягнули випадка, коли поточний масив містить лише два елементи. Тоді ми так само викликаємо процедуру розділення з медіаною, просто в якості середнього автоматично буде обраний перший елемент масиву.

Вхідні дані для цього завдання такі самі, що й для завдання №1.

Введіть загальну кількість порівнянь елементів в процедурі Partition:

Перевірка

Зберегти

Показати відповідь

Ви використали 2 з 5 можливостей надіслати свої матеріали на розгляд.



[Про нас](#) [Преса](#) [FAQ](#) [Контакти](#)

© 2015 Prometheus, some rights reserved

- [Умови надання послуг](#) та [Кодекс Честі](#)

