

Национальный исследовательский Университет ИТМО
Мегафакультет информационных и трансляционных технологий
Факультет инфокоммуникационных технологий

Алгоритмы и структуры данных

Лабораторная работа №5

Работу

выполнил:

А.И. Либерман

Группа: К3124

Преподаватель:

В.Е. Артамонова

Санкт-Петербург
2022

Содержание

1. Тема лабораторной работы	3
2. Определение	3
3. Задача №1: Неубывающая пирамида	3
4. Решение	4
4.1. Листинг кода	4
4.2. Текстовое объяснение решения задачи	4
4.3. Результат работы кода на примере из текста задачи	4
Вывод	5
Список использованных источников	6

1. Тема лабораторной работы

Лабораторная работа посвящена разбору следующих структур данных: деревья, пирамида или двоичная куча, очередь с приоритетами, а также пирамидальной сортировке - еще одному виду сортировки за время (1)

$$T(n) = T1(n) + T2(n) = O(n \log(n)). \quad (1)$$

2. Определение

Неубывающая пирамида - это почти полное дерево (только уровень листьев может быть неполным), удовлетворяющее требованию – ключ каждой вершины не больше ключа родителя. Аналогично определяются невозрастающие пирамиды.

[1; 2]

3. Задача №1: Неубывающая пирамида

Структуру данных «куча», или, более конкретно, «неубывающая пирамида», можно реализовать на основе массива. Для этого должно выполняться основное свойство неубывающей пирамиды, которое заключается в том, что для каждого $1 \leq i \leq n$ выполняются условия:

- если $2i \leq n$, то $a_i \leq a_{2i}$,
- если $2i + 1 \leq n$, то $a_i \leq a_{2i+1}$.

Дан массив целых чисел. Определите, является ли он неубывающей пирамидой.

- **Формат входного файла (input.txt).** Первая строка входного файла содержит целое число n ($1 \leq n \leq 106$). Вторая строка содержит n целых чисел, по модулю не превосходящих $2 * 109$.
- **Формат выходного файла (output.txt).** Выведите «YES», если массив является неубывающей пирамидой, и «NO» в противном случае.
- Ограничение по времени. 2 сек.
- Ограничение по памяти. 256 мб.

Таблица 3.1

Примеры входных и выходных файлов

№	input.txt	output.txt
1	1 0 1 2 0	NO
2	1 3 2 5 4	YES

4. Решение

4.1. Листинг кода

```
1 def checkMinHeap(A, i):
2     if 2 * i + 2 > len(A):
3         return True
4     left = (A[i] ≤ A[2 * i + 1]) and checkMinHeap(A, 2 * i + 1)
5     right = (2 * i + 2 == len(A)) or (A[i] ≤ A[2 * i + 2] and
6     ↪ checkMinHeap(A, 2 * i + 2))
7     return left and right
8
9 f = open("input.txt", "r")
10 d = open("output.txt", "w")
11 n = int(f.readline())
12 a = list(f.readline().split())
13 i = 0
14 if checkMinHeap(a, i):
15     d.write(str("YES"))
16 else:
17     d.write(str("NO"))
```

Листинг 1: Код задачи №1

4.2. Текстовое объяснение решения задачи

Данная задача позволяет на самом простом примере разобраться с таким типом данных как куча или же неубывающая пирамида. При помощи функции необходимо проверить верность равенства, данного в тексте задачи и вывести в output файл слова “YES” или “NO” в зависимости от результата выполнения кода.

4.3. Результат работы кода на примере из текста задачи

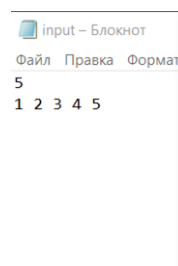


Рисунок 4.1. Input file

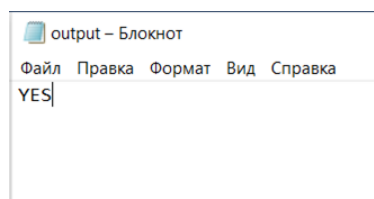


Рисунок 4.2. Output file

Вывод

в лабораторной работе я встретила с понятиями кучи и очереди с приоритетами, разобралась в их взаимосвязи и рассмотрела разные варианты реализации подобных кодов.

Список использованных источников

1. *Dasgupta S., Papadimitriou C. H., Vazirani U. V.* Algorithms. — McGraw-Hill Higher Education New York, 2008.
2. Introduction to algorithms / Т. Н. Cormen [и др.]. — MIT press, 2022.